Linearised implicit time-advancing applied to sediment transport simulations

Marco Bilanceri, François Beux, Imad Elmahi, Hervé Guillard, Maria Vittoria Salvetti

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Linearised implicit time-advancing applied to sediment transport simulations

Marco Bilanceri, François Beux ,Imad Elmahi ,Hervé Guillard, Maria Vittoria Salvetti

N° 7492

December 16, 2010

Thème NUM

apport de recherche

ISSN 0249-6399 ISRN INRIA/RR--7492--FR+ENG

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

INRIA

centre de recherche
SOPHIA ANTIPOLIS - MÉDITERRANÉE

Linearised implicit time-advancing applied to sediment transport simulations

Marco Bilanceri*, François Beux† ,Imad Elmahi‡ ,Hervé Guillard§, Maria Vittoria Salvetti¶

Thème NUM — Systèmes numériques
Équipe-Projet PUMAS

Rapport de recherche n° 7492 — December 16, 2010 — 44 pages

Abstract: The numerical simulation of sediment transport problems is considered. The problem is modeled through the shallow-water equations coupled with the Exner equation to describe the time evolution of the bed profile. The Grass model is used for the sediment transport. The governing equations are discretized by using two different finite-volume methods, the SRHN predictor-corrector scheme and a Modified Roe scheme for non conservative systems of equations. As for the time advancing, starting from the explicit versions, linearised implicit schemes are generated, in which the flux Jacobians are computed through automatic differentiation. This allows the complexity of the analytical differentiation of the numerical schemes to be avoided. Second-order accuracy in space and time is obtained through MUSCL reconstruction together with a defect-correction approach. Finally the considered numerical ingredients are compared in terms of accuracy and computational time using different one dimensional and two dimensional sediment transport problems, characterised by different time scales for the evolution of the bed and of the water flow.

Key-words: CFD, Numerical analysis, Implicit schemes, Defect Correction, Exner, St Venant, Sediment transport, Finite Volume

* Dip. Ingegneria Aerospaziale, Università di Pisa, Via G. Caruso 8, 56122 Pisa (Italy), e-mail: marco.bilanceri@ing.unipi.it

† Alta S.p.A.,Via Gherardesca, 5 - 56121 Ospedaletto, Pisa (Italy), e-mail: f.beux@alta-space.com

‡ EMCS, Ensa, Oujda, Complexe Universitaire, BP 669, 60000 Oujda (Marocco), e-mail: ielmahi@ensa.ump.ma

§ INRIA, BP 93, 06902 Sophia Antipolis Cedex (France) and Laboratoire Jean-Alexandre Dieudonné, University of Nice Sophia-Antipolis, Parc Valrose 06108 Nice Cedex (France) : e-mail: Herve.Guillard@inria.fr

¶ Dip. Ingegneria Aerospaziale, Università di Pisa, Via G. Caruso 8, 56122 Pisa (Italy), e-mail: mv.salvetti@ing.unipi.it

Centre de recherche INRIA Sophia Antipolis – Méditerranée
2004, route des Lucioles, BP 93, 06902 Sophia Antipolis Cedex
Téléphone : +33 4 92 38 77 77 — Télécopie : +33 4 92 38 77 65

Schéma temporelle implicite linéarisé appliqué à des simulations de transport de sédiments

Résumé : On considère dans ce travail, la simulation numérique d'un modèle de transport de sédiment constitué des équations de St Venant couplées à une équation d'Exner qui décrit l'évolution temporelle d'un fond mobile. Le modèle de Grass est utilisé pour décrire le transport de sédiment. Ce système d'équation est approché en utilisant deux schémas de volumes finis différents : le schéma SRHN qui est un schéma prédicteur-correcteur et un schéma de Roe modifié pour les systèmes non-conservatifs. En ce qui concerne le schéma en temps, partant d'un schéma explicite, on construit des schémas implicites linéarisés en utilisant les outils de la différentiation automatique de programmes, ce qui permet de faire l'économie de calculs analytiques complexes. Le second-ordre en temps et en espace est atteint en utilisant une technique de defect-correction. L'ensemble de ces ingrédients numériques sont comparés en terme de temps de calcul et de précision sur des cas-test uni et bi-dimensionnels caractérisés par des échelles de temps différentes pour l'évolution du fond et de la colonne d'eau.

Mots-clés : CFD, Analyse numérique, Schémas implicites, Defect Correction, Exner, St Venant, Transport de Sediment, Volumes Finis

Acknowledgments

This work has been realized in the framework of the EuroMéditerranée 3+3 network : MhyCoF. A part of it has been done while Imad Elmahi and Marco Bilanceri have visited Inria Sophia-Antipolis with the support of this program.

Contents

1 Introduction **7**

2 Physical Modeling **8**

3 Numerical Discretization **10**
3.1 General definitions for the considered finite volume formulation . 10
3.2 SRNH Scheme 12
3.2.1 Second order extension 13
3.3 Modified Roe scheme 14
3.3.1 Second order extension 15
3.4 Implicit time advancing 17
3.4.1 First-order schemes 17
3.4.2 Second-order schemes 18

4 Numerical Applications **20**
4.1 1D Numerical Experiments 20
4.1.1 Slow speed of interaction between the bed-load and water flow 21
4.1.2 Slow/Intermediate speed of interaction between bed-load and water flow 25
4.1.3 Intermediate/Fast speed of interaction between bed-load and water flow 27
4.1.4 Fast speed of interaction between bed-load and water flow 28
4.2 2D Numerical Experiments 30
4.2.1 Fast interaction between the bed-load and water flow . . . 32
4.2.2 Slow interaction between the bed-load and water flow . . 34

5 Concluding remarks **36**

A Additional details on the SRNH Scheme **39**

B Additional details on the Modified Roe Scheme **39**

C Gradient computation **41**
C.1 MR scheme 41
C.2 SRNH scheme 42

List of Figures

1 Generation of the dual Mesh. 11
2 Bottom height results computed by the SRNH scheme and $A_g = 10^{-3}$: comparison between explicit and implicit, 1^{st} and 2^{nd}-order formulations. 22
3 Bottom height results computed by the MR scheme and $A_g = 10^{-3}$: comparison between explicit and implicit, 1^{st} and 2^{nd}-order formulations. 22
4 Bottom height obtained with the SRNH implicit scheme at different values of CFL, $A_g = 10^{-3}$, 250 Cells. 23
5 Bottom height obtained with the MR implicit scheme at different values of CFL, $A_g = 10^{-3}$, 250 Cells. 23
6 Effect of the number of Defect Correction iterations on the refined grid using a CFL value of 10^4: $A_g = 10^{-3}$; (left) SRNH scheme, (right) MR scheme. 24
7 Bottom height obtained with the SRNH implicit scheme at different values of CFL, $A_g = 10^{-2}$, 250 Cells. 25
8 Bottom height obtained with the MR implicit scheme at different values of CFL, $A_g = 10^{-2}$, 250 Cells. 26
9 Bottom height results computed by the SRNH scheme and $A_g = 10^{-2}$: comparison between explicit and implicit, 1^{st} and 2^{nd}-order formulations. 26
10 Bottom height results computed by the MR scheme and $A_g = 10^{-2}$: comparison between explicit and implicit, 1^{st} and 2^{nd}-order formulations. 27
11 Bottom height results computed by the SRNH scheme and $A_g = 10^{-1}$: comparison between explicit and implicit, 1^{st} and 2^{nd}-order formulations. 28
12 Bottom height results computed by the MR scheme and $A_g = 10^{-1}$: comparison between explicit and implicit, 1^{st} and 2^{nd}-order formulations. 28
13 Bottom height results computed by the SRNH scheme and $A_g = 10^{-0}$: comparison between explicit and implicit, 1^{st} and 2^{nd}-order formulations. 29
14 Bottom height results computed by the MR scheme and $A_g = 10^{-0}$: comparison between explicit and implicit, 1^{st} and 2^{nd}-order formulations. 30
15 Ratio between the implicit computational time and the explicit one as a function of A_g. 31
16 Comparison of the results for the bed profile of the 2^{nd}-order MR scheme, $A_g = 1$, Grid GR1. 33
17 Comparison of the results for the bed profile of the 2^{nd}-order SRNH scheme, $A_g = 1$, Grid GR2. 34

18 Comparison of the effect of the value of Tol and of the number of DeC iterations for the SRNH scheme on the bed profile, CFL= 1, GR1. 35
19 Comparison of the results of the bed profile of the explicit and implicit MR scheme schemes, for 1^{st} (top) and 2^{nd}-order (bottom) of accuracy: $A_g = 10^{-3}$,Grid GR2. 36
20 Comparison of the results of the bed profile of the explicit and implicit SRNH scheme schemes, for 1^{st} (top) and 2^{nd}-order (bottom) of accuracy: $A_g = 10^{-3}$,Grid GR1. 37
21 Comparison of the results for the bed profile of the 2^{nd}-order implicit MR scheme, $A_g = 10^{-3}$, Grid GR1. 38

1 Introduction

The design and validation of numerical methods for the simulation of bedload sediment transport processes caused by the movement of a fluid in contact with the sediment layer has a significant interest for environmental and engineering problems. A few examples of such problems are beach profile changes due to severe climate waves, seabed response to dredging procedures or imposed structures, harbor siltation or transport in gravel-bed rivers.

The hydrodynamics part is usually modeled through the classical shallow-water equations coupled with an additional equation modeling the morphodynamical component. This last equation is usually a continuity or Exner equation, expressing the conservation of the sediment volume, in which the solid transport discharge is provided by a closure model. Many different models of solid transport discharge are available in the literature (see, e.g., [6] for a review). As a first step, the Grass equation [11] is considered herein, which is one of the most popular and simple models.

A huge amount of work has been done in the last decades to develop numerical methods for the simulation of the previous system of equations (see, e.g., the references in [6, 8, 3, 2]). The treatment of source terms and of the bed-load fluxes has received the largest attention. Indeed, a well known problem is that shallow water equations on non-flat topography have steady-state solutions in which the flux gradients are non-zero but are exactly balanced by the source terms. Standard numerical methods for the discretization of conservation laws may fail in correctly reproducing this balance (C-property, see e.g. [5]).

On the other hand, in shallow-water problems, time advancing has received much less attention and it is usually carried out by explicit schemes. The focus of the present paper is on the comparison between explicit and implicit schemes in the simulation of coupled shallow-water equations and sediment transport. Indeed, if the interaction of the water flow with the mobile bed is weak, the characteristic time scales of the flow and of the sediment transport can be very different introducing time stiffness in the global problem. For these cases, the stability properties of explicit schemes may significantly be deteriorated and, hence, it can be advantageous to use implicit schemes. Implicit schemes might also be useful if morphodynamic models more complex than the Exner/Grass one, which lead to a more stiff evolution of the bed (see e.g. [6]), are used. On the other hand, since the considered problems are unsteady, attention must be paid for implicit schemes in the choice of the time step. Indeed, a too large time step could deteriorate the accuracy of the results and one issue is to investigate whether and for which conditions the use of implicit schemes is really convenient from a computational viewpoint. A first investigation of this issue is provided in the present paper for 1D and 2D sediment-transport problems, involving different rates of bedload/water-flow interaction. Another difficulty with implicit schemes is that, in order to avoid the solution of a nonlinear system at each time step, the numerical fluxes must be linearised in time and this is classically done via differentiation by computing the Jacobian of the fluxes with respect to the flow variables. Nevertheless, it is not always possible nor

convenient to exactly compute the Jacobian matrices, because it is not unusual to have some lack of differentiability of the numerical flux functions or the computation may be complex for some schemes, as e.g. those involving projector-corrector stages. In order to overcome these difficulties, we use an automatic differentiation tool (Tapenade, [12], http://www-sop.inria.fr/tropics/). A defect-correction approach [14], which consists in iteratively solving linear systems involving the 1st-order flux Jacobians, is finally used to obtain second-order accuracy (both in time and space) at limited computational costs. Note that automatic differentiation together with the defect-correct approach allow the numerical method to be easily adapted to changes in the physical model, such as for instance the use of different models for the solid transport discharge.

2 Physical Modeling

The physical model used in the present study consists in the well known shallow-water equations coupled with an additional equation to describe the transport of sediment. Neglecting wind effects, Coriolis forces and friction losses, the 2D shallow water equations may be written in the following conservative form:

$$\begin{aligned} &\frac{\partial h}{\partial t} + \frac{\partial hu}{\partial x} + \frac{\partial hv}{\partial y} = 0 \\ &\frac{\partial hu}{\partial t} + \frac{\partial}{\partial x}\left(hu^2 + \frac{1}{2}gh^2\right) + \frac{\partial}{\partial y}(huv) = -gh\frac{\partial Z}{\partial x} \\ &\frac{\partial hv}{\partial t} + \frac{\partial}{\partial x}(huv) + \frac{\partial}{\partial y}\left(hv^2 + \frac{1}{2}gh^2\right) = -gh\frac{\partial Z}{\partial y} \end{aligned} \tag{1}$$

where x and y are the spatial coordinates, t is the time, h is the height of the flow above the bottom Z, g is acceleration of gravity and u and v are the velocity components in the x and y directions. In the standard shallow-water formulation the bottom is a function of space only, that is $Z = Z(x, y)$. To include the effect of sediment transport, an additional equation which describes the time evolution of the bed level is required. The Exner equation, a well-known and a common choice for this kind of problems, has been used here:

$$(1-p)\frac{\partial Z}{\partial t} + \frac{\partial Q_1}{\partial x} + \frac{\partial Q_2}{\partial y} = 0 \tag{2}$$

where p is the (constant) sediment porosity and Q_1 and Q_2 are the bed-load sediment transport fluxes in the x and y directions. In this study the sediment transport fluxes are defined through the use of the Grass model [11]:

$$Q_1 = A_g u\left(u^2+v^2\right)^{\frac{m-1}{2}}, \qquad Q_2 = A_g v\left(u^2+v^2\right)^{\frac{m-1}{2}} \tag{3}$$

where A_g and $1 \leq m \leq 4$ are experimental constants depending on the particular problem under consideration. More precisely, in the following only the value

$m = 3$ is considered so that (3) reduces to:

$$Q_1 = A_g u \left(u^2 + v^2\right), \qquad Q_2 = A_g v \left(u^2 + v^2\right) \tag{4}$$

Note that instead of the classical model (4), used here for the sake of simplicity, more complex formulations can also be considered for the formulation of Q_1 and Q_2 (see e.g.[6]). Using (1) and (2) the following complete system is then obtained:

$$\left\{ \begin{array}{lcr} \dfrac{\partial h}{\partial t} + \dfrac{\partial hu}{\partial x} + \dfrac{\partial hv}{\partial y} & = & 0 \\ \dfrac{\partial hu}{\partial t} + \dfrac{\partial}{\partial x}\left(hu^2 + \dfrac{1}{2}gh^2\right) + \dfrac{\partial}{\partial y}(huv) & = & -gh\dfrac{\partial Z}{\partial x} \\ \dfrac{\partial hv}{\partial t} + \dfrac{\partial}{\partial x}(huv) + \dfrac{\partial}{\partial y}\left(hv^2 + \dfrac{1}{2}gh^2\right) & = & -gh\dfrac{\partial Z}{\partial y} \\ \dfrac{\partial Z}{\partial t} + \xi\dfrac{\partial Q_1}{\partial x} + \xi\dfrac{\partial Q_2}{\partial y} & = & 0 \end{array} \right. \tag{5}$$

where $\xi = \frac{1}{1-p}$. It is possible to write (5) as a system of conservation laws with a source term, that is:

$$\frac{\partial \mathbf{W}}{\partial t} + \frac{\partial \mathbf{F_1}(\mathbf{W})}{\partial x} + \frac{\partial \mathbf{F_2}(\mathbf{W})}{\partial y} = \mathbf{S}(\mathbf{W}) \tag{6}$$

where

$$\left\{ \begin{array}{lccccc} \mathbf{W} & = (& h, & hu, & hv, & Z \quad)^T \\ \mathbf{F_1}(\mathbf{W}) & = (& hu, & hu^2 + \dfrac{1}{2}gh^2, & huv, & \xi Q_1 \quad)^T \\ \mathbf{F_2}(\mathbf{W}) & = (& hv, & hvu, & hv^2 + \dfrac{1}{2}gh^2, & \xi Q_2 \quad)^T \\ \mathbf{S}(\mathbf{W}) & = (& 0, & -gh\dfrac{\partial Z}{\partial x}, & -gh\dfrac{\partial Z}{\partial y}, & 0 \quad)^T \end{array} \right. \tag{7}$$

The Jacobian matrix of $\mathbf{F_1}(\mathbf{W})$ and $\mathbf{F_2}(\mathbf{W})$ in (7) is singular: to avoid this singularity, it is possible to incorporate the variable Z into the flux function [13]. The resulting system is:

$$\frac{\partial \mathbf{W}}{\partial t} + \frac{\partial \tilde{\mathbf{F}}_\mathbf{1}(\mathbf{W})}{\partial x} + \frac{\partial \tilde{\mathbf{F}}_\mathbf{2}(\mathbf{W})}{\partial y} = \tilde{\mathbf{S}}(\mathbf{W}) \tag{8}$$

where

$$\left\{ \begin{array}{lccccc} \tilde{\mathbf{F}}_\mathbf{1}(\mathbf{W}) & = (& hu, & hu^2 + \dfrac{1}{2}gh^2 + ghZ, & huv, & \xi Q_1 \quad)^T \\ \tilde{\mathbf{F}}_\mathbf{2}(\mathbf{W}) & = (& hv, & hvu, & hv^2 + \dfrac{1}{2}gh^2 + ghZ, & \xi Q_2 \quad)^T \\ \tilde{\mathbf{S}}(\tilde{\mathbf{W}}) & = (& 0, & gZ\dfrac{\partial h}{\partial x}, & gZ\dfrac{\partial h}{\partial y}, & 0 \quad)^T \end{array} \right. \tag{9}$$

Alternatively, following the work in [6, 8], it is possible to consider (5) as a non-conservative system in the form:

$$\frac{\partial \hat{\mathbf{W}}}{\partial t} + \frac{\partial \hat{\mathbf{F}}_{\mathbf{1}}(\hat{\mathbf{W}})}{\partial x} + \frac{\partial \hat{\mathbf{F}}_{\mathbf{2}}(\hat{\mathbf{W}})}{\partial y} = \hat{\mathbf{B}}_{\mathbf{1}}(\hat{\mathbf{W}})\frac{\partial \hat{\mathbf{W}}}{\partial x} + \hat{\mathbf{B}}_{\mathbf{2}}(\hat{\mathbf{W}})\frac{\partial \hat{\mathbf{W}}}{\partial y} \tag{10}$$

where

$$\left\{ \begin{array}{lccccc} \hat{\mathbf{W}} & = (& h, & hu, & hv, & H \quad)^T \\ \hat{\mathbf{F}}_{\mathbf{1}}(\hat{\mathbf{W}}) & = (& hu, & hu^2 + \frac{1}{2}gh^2, & huv, & -\xi Q_1 \quad)^T \\ \hat{\mathbf{F}}_{\mathbf{2}}(\hat{\mathbf{W}}) & = (& hv, & hvu, & hv^2 + \frac{1}{2}gh^2, & -\xi Q_2 \quad)^T \end{array} \right. \tag{11}$$

and

$$\hat{\mathbf{B}}_{\mathbf{1}}(\hat{\mathbf{W}}) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & gh \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad \hat{\mathbf{B}}_{\mathbf{2}}(\hat{\mathbf{W}}) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & gh \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{12}$$

Note that, following [6, 8], in (11) the variable Z is replaced by:

$$H = \hat{H} - Z$$

where $\hat{H}$ is a fixed reference level.

3 Numerical Discretization

In this section the numerical discretization of the problem presented in section 2 is addressed. First some general definitions for the considered finite volume discretization are provided. After that, the numerical formulation of two explicit numerical methods, the SRNH scheme and the Modified Roe one (MR in the following) is presented. Finally, at the end of this section the generation of an implicit scheme starting from its explicit counterpart is considered.

3.1 General definitions for the considered finite volume formulation

The considered space discretization is based on a finite volume approach. At a preliminary stage, the considered 2D computational domain $\mathcal{V} \in \mathbb{R}^2$ is approximated by means of a polygonal domain $\mathcal{V}^{pol}$ which, in turn, is divided into N_t triangles having vertexes $\mathbf{P}_i$, with $i \in \mathcal{I} := \{1, \ldots, N_c\}$. Let T_h, with $h \in \mathcal{H} := \{1, \ldots, N_t\}$, denote the h^{th}-triangle: the i^{th} finite volume cell V_i, associated with $\mathbf{P}_i$, is given by:

$$V_i = \bigcup_{h \in t(i)} V_i^{(h)}$$

where:

- $t(i)$ is the set of indexes marking those triangles which share $\mathbf{P}_i$ as a vertex;
- $V_r^{(h)}$ represents the subset of T_h which is defined by further dividing T_h into six sub-triangles by means of its medians and subsequently considering those two sub-triangles which share $\mathbf{P}_r$ as a vertex.

Clearly, there is a finite volume cell for each vertex. The considered finite volume discretization is sometimes referred to as a "dual mesh" (see e.g. [9]), by virtue of the specific procedure which is adopted in order to build the cells starting from the triangles.
The following notation is considered: given a finite volume V_i, $|V_i|$ is its area and $\mathbf{G}_i$ is its centre of mass. $N(i)$ is the set of neighbouring cells of the i^{th}-cell, $\mathcal{B}_i = N(i) \cup \{i\}$ and the set $\overline{\mathcal{B}}_i = \bigcup_{j \in \mathcal{B}_i} \mathcal{B}_j$ contains the i^{th}-cell, its neighbours and the neighbours of the neighbours.
$\Gamma_{ij,1}$ and $\Gamma_{ij,2}$ are the two segments of the common interface between cell V_i and V_j and $|\Gamma_{ij,1}|$ and $|\Gamma_{ij,2}|$ their length (see figure 1). The normal unit vector to $\Gamma_{ij,1}$ is $\mathbf{n}_{ij,1} = (n_{x,ij,1}, n_{y,ij,1})$ and a similar definition holds for $\mathbf{n}_{ij,2}$. The

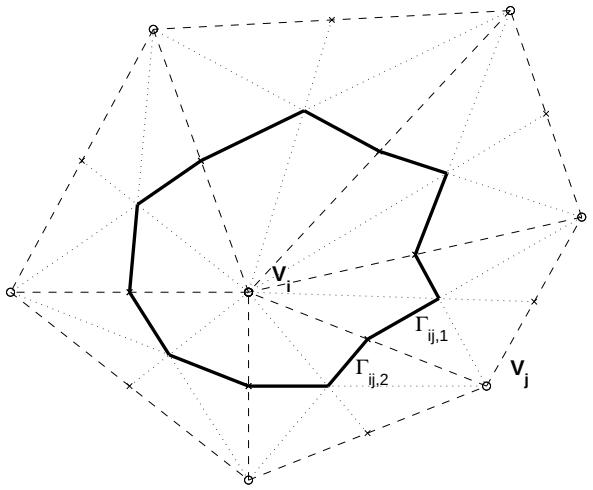

Figure 1: Generation of the dual Mesh.

average normal vector for the interface between the i^{th} and j^{th}-cell is defined as follows:

$$|\Gamma_{ij}|\mathbf{n}_{ij} = |\Gamma_{ij,1}|\mathbf{n}_{ij,1} + |\Gamma_{ij,2}|\mathbf{n}_{ij,2}$$

where $\mathbf{n}_{ij}$ is the normal unit vector and $|\Gamma_{ij}|$ is its length. Finally $\mathbf{W}_i^n$ is the average value of the solution $\mathbf{W}$ in the i^{th} cell at time t_n:

$$\mathbf{W}_i^n \simeq \frac{1}{|V_i|} \int_{V_i} \mathbf{W}(\mathbf{x}, t_n) dV \tag{13}$$

3.2 SRNH Scheme

The SRNH numerical scheme is a numerical method tested and developed in several papers, [1, 3] among others. Herein the main features of the explicit scheme are quickly outlined, for additional details we refer to [3]. Note that the 2D formulation presented in this work is slightly different from the original formulation in [3]: in the original paper the spatial discretization for the system of equations of the sediment transport problem is based on the formulation (6)-(7). In this study we considered the same numerical method based on a different formulation of the system of equations, namely (8)-(9). This approach has been developed and tested in the 1D case in [4] but, at least in the authors knowledge, the 2D extension has never been studied. Integrating (8) over a control volume V_i the resulting system of equations is:

$$\frac{\partial \mathbf{W}_i}{\partial t} = -\frac{1}{|V_i|}\sum_{j\in N(i)}\int_{\Gamma_{ij}} \tilde{\mathcal{F}}(\mathbf{W},\mathbf{n})d\sigma + \frac{1}{|V_i|}\int_{V_i}\tilde{\mathbf{S}}(\mathbf{W})dV \tag{14}$$

where

$$\tilde{\mathcal{F}}(\mathbf{W},\mathbf{n}) \doteq n_x\tilde{\mathbf{F}}_1(\mathbf{W}) + n_y\tilde{\mathbf{F}}_2(\mathbf{W}) \tag{15}$$

To discretize (14) it is first necessary to define a numerical approximation for the flux function $\tilde{\mathcal{F}}$ and for the source term $\tilde{\mathbf{S}}$. The SRNH scheme [3] is composed by a predictor and a corrector stage: in the predictor stage an averaged state $\mathbf{W}_{ij}^n$ is computed, then this predicted state is used in the corrector stage to update the solution. In particular in the SRNH scheme the flux at the interface Γ_{ij} is approximated with the analytical flux function computed using the predicted state $\mathbf{W}_{ij}^n$:

$$\int_{\Gamma_{ij}} \tilde{\mathcal{F}}(\mathbf{W},\mathbf{n},t^n)d\sigma \simeq \tilde{\mathcal{F}}(\mathbf{W}_{ij}^n,\mathbf{n}_{ij})|\Gamma_{ij}| \tag{16}$$

The predictor stage is based on primitive variables projected on the normal and tangential directions with respect to the cell interface (η and τ respectively). Defining the vector $\mathbf{U}$ of the primitive projected variables as:

$$\mathbf{U} = \begin{pmatrix} h, & u_\eta, & u_\tau, & Z \end{pmatrix} \tag{17}$$

where

$$u_\eta = un_x + vn_y, \quad u_\tau = -un_y + vn_x \tag{18}$$

The predictor step of the SRNH scheme is then computed as follows:

$$\mathbf{U}_{ij}^n = \frac{1}{2}\left(\mathbf{U}_i^n + \mathbf{U}_j^n\right) - \frac{1}{2}\mathrm{sgn}\left[\mathbf{A}_\eta(\overline{\mathbf{U}}_{ij}^n)\right]\left(\mathbf{U}_j^n - \mathbf{U}_i^n\right) \tag{19}$$

where $\overline{\mathbf{U}}_{ij}^n = \overline{\mathbf{U}}\left(\mathbf{U}_i^n, \mathbf{U}_j^n\right)$ is a suitable Roe averaged state and $\mathrm{sgn}\left[\mathbf{A}_\eta(\mathbf{U})\right]$ is the sign matrix associated to the matrix $\mathbf{A}_\eta(\mathbf{U})$: the expressions for $\overline{\mathbf{U}}$, $\mathrm{sgn}\left[\mathbf{A}_\eta(\mathbf{U})\right]$ and $\mathbf{A}_\eta(\mathbf{U})$ are reported in appendix A. Once $\mathbf{U}_{ij}^n$ is available, the state $\mathbf{W}_{ij}^n$ appearing in (16) is simply $\mathbf{U}_{ij}^n$ expressed in conservative variables.

Finally it is necessary to define an approximation of the source term such that:

$$\frac{1}{|V_i|}\int_{V_i}\tilde{\mathbf{S}}(\mathbf{W},t^n)dV \simeq \tilde{\mathbf{S}}_i^n \tag{20}$$

In [3], starting from the system (6)-(7) the discretization of the source term is defined so that the resulting numerical scheme is well-balanced and satisfy the C-property [5]. Applying the same approach to the system (8)-(9) it is possible to demonstrate that the following discretization of the source term defines a well-balanced scheme:

$$\tilde{\mathbf{S}}_i^n = \tilde{\mathbf{S}}_i\left(\{\mathbf{W}_{ij}^n\}_{j\in N(i)}\right)$$

where

$$\left\{\begin{array}{l} \overline{Z}_{x,i} = \dfrac{1}{2}\dfrac{\displaystyle\sum_{j\in N(i)}\left(Z_{ij}\right)^2 n_{x,ij}|\Gamma_{ij}|}{\displaystyle\sum_{j\in N(i)} Z_{ij}n_{x,ij}|\Gamma_{ij}|}, \quad \overline{Z}_{y,i} = \dfrac{1}{2}\dfrac{\displaystyle\sum_{j\in N(i)}\left(Z_{ij}\right)^2 n_{y,ij}|\Gamma_{ij}|}{\displaystyle\sum_{j\in N(i)} Z_{ij}n_{y,ij}|\Gamma_{ij}|} \\ \\ \tilde{\mathbf{S}}_i\left(\{\mathbf{W}_{ij}\}_{j\in N(i)}\right) = \left(0, \quad g\overline{Z}_{x,i}\displaystyle\sum_{j\in N(i)} h_{ij}n_{x,ij}|\Gamma_{ij}|, \quad g\overline{Z}_{y,i}\sum_{j\in N(i)} h_{ij}n_{y,ij}|\Gamma_{ij}|, \quad 0\right)^T \end{array}\right. \tag{21}$$

By injecting (16) and (20) in (14), the following semi-discrete form of the system is obtained:

$$\frac{\partial \mathbf{W}_i}{\partial t} = \mathrm{RHS}_1\left(\{\mathbf{W}_j^n\}_{j\in\mathcal{B}_i}\right) \tag{22}$$

where

$$\mathrm{RHS}_1\left(\{\mathbf{W}_j^n\}_{j\in\mathcal{B}_i}\right) = -\frac{1}{|V_i|}\sum_{j\in N(i)}\tilde{\mathcal{F}}(\mathbf{W}_{ij}^n,\mathbf{n}_{ij})|\Gamma_{ij}| + \tilde{\mathbf{S}}_i^n \tag{23}$$

Finally, for the time discretization an explicit first order Euler step can be considered as in [3]. The resulting numerical scheme is:

$$\mathbf{W}_i^{n+1} = \mathbf{W}_i^n + \Delta t^n \mathrm{RHS}_1\left(\{\mathbf{W}_j^n\}_{j\in\mathcal{B}_i}\right) \tag{24}$$

3.2.1 Second order extension

The extension to second-order accuracy in space can be achieved by using a classical MUSCL technique [16], in which the flux function is computed by using the extrapolated variable values at the cell interface. First, at each interface Γ_{ij}, the extrapolated values of the variables are computed:

$$\mathbf{W}_{ij}^{n,-} = \mathbf{W}_i^n + \frac{1}{2}\nabla\mathbf{W}_i\cdot\mathbf{d}_{ij}, \quad \mathbf{W}_{ij}^{n,+} = \mathbf{W}_j^n - \frac{1}{2}\nabla\mathbf{W}_j\cdot\mathbf{d}_{ij} \tag{25}$$

where $\mathbf{d}_{ij}$ is the vector joining the node i with the node j and $\nabla\mathbf{W}_i$ is the cell gradient evaluated as shown in appendix C. The 2^{nd}-order accuracy in space

is then achieved computing the flux function and the source term using the extrapolated values (25), in analogy with the first-order scheme:

$$\left\{\begin{array}{lcl} \mathbf{U}_{ij}^{n,\pm} & = & \frac{1}{2}\left(\mathbf{U}_{ij}^{n,-}+\mathbf{U}_{ij}^{n,+}\right)-\frac{1}{2}\mathrm{sgn}\left[\mathbf{A}_\eta(\overline{\mathbf{U}}_{ij}^{n,\pm})\right]\left(\mathbf{U}_{ij}^{n,+}-\mathbf{U}_{ij}^{n,-}\right) \\ \mathrm{RHS}_2\left(\{\mathbf{W}_j^n\}_{j\in\overline{\mathcal{B}}_i}\right) & = & -\frac{1}{|V_i|}\sum_{j\in N(i)}\tilde{\mathcal{F}}(\mathbf{W}_{ij}^{n,\pm},\mathbf{n}_{ij})|\Gamma_{ij}|+\tilde{\mathbf{S}}_i^{n,\pm} \end{array}\right. \tag{26}$$

where $\overline{\mathbf{U}}_{ij}^{n,\pm}=\overline{\mathbf{U}}\left(\mathbf{U}_{ij}^{n,-},\mathbf{U}_{ij}^{n,+}\right)$ and $\tilde{\mathbf{S}}_i^{n,\pm}=\tilde{\mathbf{S}}_i\left(\{\mathbf{W}_{ij}^{n,\pm}\}_{j\in N(i)}\right)$. It is worth notice the difference between the second of (26) and (23). The set of indexes j from which the solution W_i is dependent (the stencil) is larger for the 2^{nd}-order accurate flux function than for the first order one. This is a well known feature of a second order scheme with MUSCL reconstruction and in section 3.4 we discuss its consequences for the second order implicit scheme. Using (26), the semi-discrete form of (14) becomes

$$\frac{\partial\mathbf{W}_i}{\partial t}=\mathrm{RHS}_2\left(\{\mathbf{W}_j^n\}_{j\in\overline{\mathcal{B}}_i}\right) \tag{27}$$

Finally, the second order accuracy in time can be achieved through a two-step Runge-Kutta scheme for time advancing:

$$\left\{\begin{array}{lcl} \mathbf{W}_i^{n+1/2} & = & \mathbf{W}_i^n+\Delta t^n\mathrm{RHS}_2\left(\{\mathbf{W}_j^n\}_{j\in\overline{\mathcal{B}}_i}\right) \\ \mathbf{W}_i^{n+1} & = & \frac{\mathbf{W}_i^{n+1/2}+\mathbf{W}_i^n}{2}+\frac{1}{2}\Delta t^n\mathrm{RHS}_2\left(\{\mathbf{W}_j^{n+1/2}\}_{j\in\overline{\mathcal{B}}_i}\right) \end{array}\right. \tag{28}$$

3.3 Modified Roe scheme

The following numerical method has been proposed by Castro et al. in a series of papers [6, 7, 8]. The Modified Roe scheme is based on the non conservative formulation (10) re-casted in the following form:

$$\frac{\partial\hat{\mathbf{W}}}{\partial t}+\hat{\mathbf{A}}_1(\hat{\mathbf{W}})\frac{\partial\hat{\mathbf{W}}}{\partial x}+\hat{\mathbf{A}}_2(\hat{\mathbf{W}})\frac{\partial\hat{\mathbf{W}}}{\partial y}=0 \tag{29}$$

where

$$\hat{\mathbf{A}}_k(\hat{\mathbf{W}})=\frac{\partial\hat{\mathbf{F}}_k(\hat{\mathbf{W}})}{\partial\hat{\mathbf{W}}}-\hat{\mathbf{B}}_k(\hat{\mathbf{W}})\quad k=1,2 \tag{30}$$

Note that the matrix $\hat{A}_k$ is the sum of a conservative term, the Jacobian of $\hat{\mathbf{F}}_k(\hat{\mathbf{W}})$, and a non conservative one, $\hat{\mathbf{B}}_k(\hat{\mathbf{W}})$, which takes into account the spatial variation of the bed. In [8] it is shown that it is possible to define a Roe-type scheme for the non conservative system (29) leading to the following semi-discrete form:

$$\frac{\partial\hat{\mathbf{W}}_i^n}{\partial t}=\widehat{\mathrm{RHS}}_1\left(\{\hat{\mathbf{W}}_j^n\}_{j\in\mathcal{B}_i}\right) \tag{31}$$

where

$$\widehat{\mathrm{RHS}}_1\left(\{\hat{\mathbf{W}}_j^n\}_{j\in\mathcal{B}_i}\right) = -\frac{1}{|V_i|}\sum_{j\in N(i)}|\Gamma_{ij}|\left(\hat{\mathcal{F}}(\hat{\mathbf{W}}_i^n,\hat{\mathbf{W}}_j^n,\mathbf{n}_{ij}) - \frac{1}{2}\hat{\mathbf{B}}_{ij}\left(\hat{\mathbf{W}}_j^n - \hat{\mathbf{W}}_i^n\right)\right) \tag{32}$$

and $\hat{\mathcal{F}}(\hat{\mathbf{W}}_j^n,\hat{\mathbf{W}}_i^n,\mathbf{n}_{ij})$ is a Roe-like numerical flux function defined as:

$$\hat{\mathcal{F}}(\hat{\mathbf{W}}_i^n,\hat{\mathbf{W}}_j^n,\mathbf{n}_{ij}) = \frac{1}{2}\left(\hat{\mathbf{F}}_{\mathbf{n}_{ij}}(\hat{\mathbf{W}}_i) + \hat{\mathbf{F}}_{\mathbf{n}_{ij}}(\hat{\mathbf{W}}_j)\right) - \frac{1}{2}|\hat{\mathbf{A}}_{ij}|(\hat{\mathbf{W}}_j^n,\hat{\mathbf{W}}_i^n,\mathbf{n}_{ij})\left(\hat{\mathbf{W}}_j^n - \hat{\mathbf{W}}_i^n\right) \tag{33}$$

and the expressions of $|\hat{\mathbf{A}}_{ij}|$, $\hat{\mathbf{F}}_{\mathbf{n}_{ij}}$, $\hat{\mathbf{B}}_{ij}$ are reported in appendix B. Finally, to obtain an explicit scheme first order accurate in time, the time-discretization of the scheme can be carried out using an Euler step so that the resulting numerical scheme is:

$$\mathbf{W}_i^{n+1} = \mathbf{W}_i^n + \Delta t^n \widehat{\mathrm{RHS}}_1\left(\{\mathbf{W}_j^n\}_{j\in\mathcal{B}_i}\right) \tag{34}$$

3.3.1 Second order extension

The formulation of the 2^{nd} order Modified Roe scheme follows a different approach from the one of the SRNH scheme.
First, a reconstruction operator $P_i(\mathbf{x})$ is defined at every cell. The reconstruction operator is a function of the values of the solution in the neighbour cells of the V_i, that is:

$$P_i(\mathbf{x},t^n) = P_i(\mathbf{x},\{\hat{\mathbf{W}}_j^n\}_{j\in\mathcal{B}_i})$$

In particular in this work we consider a MUSCL-like reconstruction operator

$$P_i(\mathbf{x},t^n) = \hat{\mathbf{W}}_i^n + \widehat{\nabla\hat{\mathbf{W}}_i^n}\,(\mathbf{x}-\mathbf{G_i}) \tag{35}$$

where $\widehat{\nabla\hat{\mathbf{W}}_i^n}$ is an approximation of the gradient in the i^{th}-cell, possibly taking into account flux-limiters, and is defined in appendix C. Once the reconstruction operator (35) is defined, it is possible to extend the Modified Roe scheme to the second order accuracy. The semi-discrete formulation of (29) is:

$$\begin{aligned}\frac{\partial\hat{\mathbf{W}}}{\partial t} = -\frac{1}{|V_i|}\Bigg(&\sum_{j\in N(i)}\int_{\Gamma_{ij}}\Big(\hat{\mathcal{F}}\left(\hat{\mathbf{W}}_{ij}^-(\sigma,t),\hat{\mathbf{W}}_{ij}^+(\sigma,t),\mathbf{n}_{ij}\right)\\ &-\frac{1}{2}\hat{\mathbf{B}}_{ij}(\sigma,t)\left(\hat{\mathbf{W}}_{ij}^+(\sigma,t)-\hat{\mathbf{W}}_{ij}^-(\sigma,t)\right)\Big)\,d\sigma\\ &-\int_{V_i}\left(\hat{\mathbf{B}}_1(P_i(\mathbf{x}))\frac{\partial P_i(\mathbf{x})}{\partial x}+\hat{\mathbf{B}}_2(P_i(\mathbf{x}))\frac{\partial P_i(\mathbf{x})}{\partial y}\right)dV\Bigg)\end{aligned} \tag{36}$$

where σ is a point of the interface Γ_{ij}, $\hat{\mathbf{W}}_{ij}^-(\sigma,t) = P_i(\sigma,t)$ and $\hat{\mathbf{W}}_{ij}^+(\sigma,t) = P_j(\sigma,t)$. Let us remark a key difference between the SRNH scheme and the Modified Roe one. In the SRNH scheme the second order extension is dependent only from the extrapolated values of the solution at the cell interfaces.

Conversely, in the Modified Roe scheme, due to the non-conservative formulation, the second order scheme is also function of the extrapolated values of the solution in the interior of the cells.
The integrals in (36) are numerically approximated and, in order to preserve the second order spatial accuracy of the scheme the order of the quadrature formula must be higher than that of the reconstruction operator. In [8] it has been shown that the third order formulas for the line integrals and the barycentre quadrature formula for the volume integrals satisfy both criteria. As a consequence it is possible to approximate the line integrals over Γ_{ij} as:

$$\int_{\Gamma_{ij}} \left(\hat{\mathcal{F}} \left(\hat{\mathbf{W}}_{ij}^{-}(\sigma), \hat{\mathbf{W}}_{ij}^{+}(\sigma), \mathbf{n}_{ij} \right) - \frac{1}{2} \hat{\mathbf{B}}_{ij}(\sigma) \left(\hat{\mathbf{W}}_{ij}^{+}(\sigma) - \hat{\mathbf{W}}_{ij}^{-}(\sigma) \right) \right) d\sigma \simeq$$
$$\sum_{l=1}^{2} |\Gamma_{ij,l}| \sum_{m=1}^{2} w_{lm} \left(\hat{\mathcal{F}} \left(\hat{\mathbf{W}}_{ij,lm}^{-}, \hat{\mathbf{W}}_{ij,lm}^{+}, \mathbf{n}_{ij,l} \right) - \frac{1}{2} \hat{\mathbf{B}}_{ij,lm} \left(\hat{\mathbf{W}}_{ij,lm}^{+} - \hat{\mathbf{W}}_{ij,lm}^{-} \right) \right) \tag{37}$$

where $\hat{\mathbf{W}}_{ij,lm}^{-} = P_i(\sigma_{lm})$, $\hat{\mathbf{W}}_{ij,lm}^{+} = P_j(\sigma_{lm})$ and w_{lm} and σ_{lm} are respectively, the weights and the points of the quadrature rule. Similarly, for the volume integral:

$$\begin{aligned} &\int_{V_i} \left(\hat{\mathbf{B}}_1(P_i(\mathbf{x})) \frac{\partial P_i(\mathbf{x})}{\partial x} + \hat{\mathbf{B}}_2(P_i(\mathbf{x})) \frac{\partial P_i(\mathbf{x})}{\partial y} \right) dV \simeq \\ &|V_i| \left(\hat{\mathbf{B}}_1(P_i(\mathbf{G}_i)) \frac{\partial P_i(\mathbf{G}_i)}{\partial x} + \hat{\mathbf{B}}_2(P_i(\mathbf{G}_i)) \frac{\partial P_i(\mathbf{G}_i)}{\partial y} \right) \end{aligned} \tag{38}$$

By injecting (37) and (38) in (36), the semi-discrete expression of the second order Modified Roe scheme is:

$$\frac{\partial \hat{\mathbf{W}}_i}{\partial t} = \widehat{\mathrm{RHS}}_2 \left(\{\hat{\mathbf{W}}_j\}_{j \in \overline{\mathcal{B}}_i} \right) \tag{39}$$

where $\widehat{\mathrm{RHS}}_2 \left(\{\hat{\mathbf{W}}_j\}_{j \in \overline{\mathcal{B}}_i} \right)$ is defined as follows

$$\begin{aligned} \widehat{\mathrm{RHS}}_2 \left(\{\hat{\mathbf{W}}_j\}_{j \in \overline{\mathcal{B}}_i} \right) = - \frac{1}{|V_i|} \Bigg(&\sum_{j \in N(i)} \sum_{l=1}^{2} |\Gamma_{ij,l}| \sum_{m=1}^{2} w_{lm} \Bigg(\hat{\mathcal{F}} \left(\hat{\mathbf{W}}_{ij,lm}^{-}, \hat{\mathbf{W}}_{ij,lm}^{+}, \mathbf{n}_{ij,l} \right) \\ &- \frac{1}{2} \hat{\mathbf{B}}_{ij,lm} \left(\hat{\mathbf{W}}_{ij,lm}^{+} - \hat{\mathbf{W}}_{ij,lm}^{-} \right) \Bigg) \\ &- |V_i| \left(\hat{\mathbf{B}}_1(P_i(\mathbf{G}_i)) \frac{\partial P_i(\mathbf{G}_i)}{\partial x} + \hat{\mathbf{B}}_2(P_i(\mathbf{G}_i)) \frac{\partial P_i(\mathbf{G}_i)}{\partial y} \right) \Bigg) \end{aligned} \tag{40}$$

To conclude, in order to obtain an explicit scheme second-order accurate in time, the time discretization can be carried out using a second-order TVD Runge-Kutta method [10]. The final expression of the second order numerical scheme

is:

$$\left\{\begin{array}{lcl} \hat{\mathbf{W}}_i^{n+1/2} & = & \hat{\mathbf{W}}_i^n + \Delta t^n \widehat{\mathrm{RHS}}_2\left(\{\hat{\mathbf{W}}_j^n\}_{j\in\overline{\mathcal{B}}_i}\right) \\ \hat{\mathbf{W}}_i^{n+1} & = & \dfrac{\hat{\mathbf{W}}_i^{n+1/2}+\hat{\mathbf{W}}_i^n}{2} + \dfrac{1}{2}\Delta t^n \widehat{\mathrm{RHS}}_2\left(\{\hat{\mathbf{W}}_j^{n+1/2}\}_{j\in\overline{\mathcal{B}}_i}\right) \end{array}\right. \tag{41}$$

3.4 Implicit time advancing

In this section the issue of generating an implicit scheme, starting from its explicit counterpart, is addressed. Initially only first-order numerical schemes are considered then the second-order extensions are discussed.

3.4.1 First-order schemes

Generally speaking, the implicit counterpart of a first order explicit Euler method is obtained considering the right hand side term as a function of the solution at time $n+1$ instead of n. That is a full implicit first order version of the schemes in equations (24) and (34) is the following:

$$\left\{\begin{array}{lcl} \mathbf{W}_i^{n+1} - \Delta t^n \mathrm{RHS}_1\left(\{\mathbf{W}_j^{n+1}\}_{j\in\mathcal{B}_i}\right) & = & \mathbf{W}_i^n \\ \hat{\mathbf{W}}_i^{n+1} - \Delta t^n \widehat{\mathrm{RHS}}_1\left(\{\hat{\mathbf{W}}_j^{n+1}\}_{j\in\mathcal{B}_i}\right) & = & \hat{\mathbf{W}}_i^n \end{array}\right. \tag{42}$$

However, from a practical point of view this would require the solution of a large non-linear system of equations at each time step. The computational cost for this operation is in general not affordable in practical applications and, in general, significantly overcomes any advantage that an implicit scheme could have with respect to its explicit counterpart. A common technique to overcome this difficulty is to linearised the numerical scheme, that is to find an approximation of $\mathrm{RHS}_1\left(\{\mathbf{W}_j^{n+1}\}_{j\in\mathcal{B}_i}\right)$ or $\widehat{\mathrm{RHS}}_1\left(\{\hat{\mathbf{W}}_j^{n+1}\}_{j\in\mathcal{B}_i}\right)$ in the form:

$$\left\{\begin{array}{lcl} \mathrm{RHS}_1\left(\{\mathbf{W}_j^{n+1}\}_{j\in\mathcal{B}_i}\right) & \simeq & \mathrm{RHS}_1\left(\{\mathbf{W}_j^{n}\}_{j\in\mathcal{B}_i}\right) + \displaystyle\sum_{j\in\mathcal{B}_i} \mathcal{D}_{ij}\Delta^n\mathbf{W}_j \\ \widehat{\mathrm{RHS}}_1\left(\{\hat{\mathbf{W}}_j^{n+1}\}_{j\in\mathcal{B}_i}\right) & \simeq & \widehat{\mathrm{RHS}}_1\left(\{\hat{\mathbf{W}}_j^{n}\}_{j\in\mathcal{B}_i}\right) + \displaystyle\sum_{j\in\mathcal{B}_i} \hat{\mathcal{D}}_{ij}\Delta^n\hat{\mathbf{W}}_j \end{array}\right. \tag{43}$$

where $\Delta^n(\,\cdot\,) = (\,\cdot\,)^{n+1} - (\,\cdot\,)^n$. Using this approximation, the following linear system must be solved at each time step for the SRNH scheme:

$$\frac{\Delta^n\mathbf{W}_i}{\Delta t^n} - \sum_{j\in\mathcal{B}_i} \mathcal{D}_{ij}\Delta^n\mathbf{W}_j = \mathrm{RHS}_1\left(\{\mathbf{W}_j^n\}_{j\in\mathcal{B}_i}\right) \tag{44}$$

and similarly for the Modified Roe scheme:

$$\frac{\Delta^n\hat{\mathbf{W}}_i}{\Delta t^n} - \sum_{j\in\mathcal{B}_i} \hat{\mathcal{D}}_{ij}\Delta^n\hat{\mathbf{W}}_j = \widehat{\mathrm{RHS}}_1\left(\{\hat{\mathbf{W}}_j^n\}_{j\in\mathcal{B}_i}\right) \tag{45}$$

The implicit linearised scheme is completely defined once a suitable definition for the matrices $\mathcal{D}_{ij}$ is given. If the right hand side is differentiable, a common choice is to use the Jacobian matrices, hence:

$$\begin{cases} \mathcal{D}_{ij} & \simeq & \left.\dfrac{\partial \mathrm{RHS}_1\left(\{\mathbf{W}_j^n\}_{j\in\mathcal{B}_i}\right)}{\partial \mathbf{W}_j}\right|_n \\ \hat{\mathcal{D}}_{ij} & \simeq & \left.\dfrac{\partial \widehat{\mathrm{RHS}}_1\left(\{\hat{\mathbf{W}}_j^n\}_{j\in\mathcal{B}_i}\right)}{\partial \hat{\mathbf{W}}_j^n}\right|_n \end{cases} \tag{46}$$

Nevertheless, it is not always possible nor convenient to exactly compute the Jacobian matrices. In fact, it is not unusual to have some lack of differentiability of the numerical flux functions or in the source term. Furthermore in the particular case of the SRNH scheme (24) the difficulty in using linearization (46) is increased by the fact that the scheme is composed by a predictor and a corrector stage. This problem has been solved herein through the use of the automatic differentiation software Tapenade [12]. The operational principle of an automatic differentiation software is as follows: given the source code of a routine which computes the function $y = F(x)$, the automatic differentiation software generates a new source code which compute the analytical derivative of the original program. In practise, each time the original program performs some operation, the differentiated program performs additional operations dealing with the differential values. For example, if the original program, at some time executes the following instruction on variables a, b, c:

$$a = b \cdot c \tag{47}$$

then the differentiated program computes also the differentials da, db, dc of these variable [12]:

$$da = db \cdot c + b \cdot dc \tag{48}$$

Through an automatic differentiation software it is possible to quickly implement an implicit linearised scheme of the form (44) or (45), once a routine which computes the explicit flux function is available. As a consequence using an automatic differentiation tool, starting from a first order explicit method, it is possible to automatically compute the matrices $\mathcal{D}_{ij}$ (or $\hat{\mathcal{D}}_{ij}$) and then implement the linearised implicit methods (44) and (45) without additional modifications.

3.4.2 Second-order schemes

A second-order implicit scheme can be obtained from its explicit counterpart using the same approach described for the first-order schemes in section 3.4.1. Therefore the second order implicit SRNH scheme would be:

$$\begin{cases} \mathbf{W}_i^{n+1/2} - \Delta t^n \mathrm{RHS}_2\left(\{\mathbf{W}_j^{n+1/2}\}_{j\in\overline{\mathcal{B}}_i}\right) & = & \mathbf{W}_i^n \\ \mathbf{W}_i^{n+1} - \dfrac{1}{2}\Delta t^n \mathrm{RHS}_2\left(\{\mathbf{W}_j^{n+1}\}_{j\in\overline{\mathcal{B}}_i}\right) & = & \dfrac{\mathbf{W}_i^{n+1/2} + \mathbf{W}_i^n}{2} \end{cases} \tag{49}$$

and similarly for the Modified Roe scheme:

$$\left\{\begin{array}{lcl} \hat{\mathbf{W}}_i^{n+1/2} - \Delta t^n \widehat{\mathrm{RHS}}_2\left(\{\hat{\mathbf{W}}_j^{n+1/2}\}_{j\in\overline{\mathcal{B}}_i}\right) & = & \hat{\mathbf{W}}_i^n \\ \hat{\mathbf{W}}_i^{n+1} - \frac{1}{2}\Delta t^n \widehat{\mathrm{RHS}}_2\left(\{\hat{\mathbf{W}}_j^{n+1}\}_{j\in\overline{\mathcal{B}}_i}\right) & = & \dfrac{\hat{\mathbf{W}}_i^{n+1/2} + \hat{\mathbf{W}}_i^n}{2} \end{array}\right. \tag{50}$$

The application of this method would require the solution of two non linear systems of equations at each time step, thus dramatically increasing the computational costs with respect to the explicit version. An alternative approach, generally more efficient in terms of computational costs, is to use a second-order backward differentiation formula in time:

$$\left\{\begin{array}{lcl} \dfrac{(1+2\tau)\mathbf{W}_i^{n+1} - (1+\tau)^2\mathbf{W}_i^n + \tau^2\mathbf{W}_i^{n-1}}{\Delta^n t\,(1+\tau)} - \mathrm{RHS}_2\left(\{\mathbf{W}_j^{n+1}\}_{j\in\overline{\mathcal{B}}_i}\right) & = & 0 \\ \dfrac{(1+2\tau)\hat{\mathbf{W}}_i^{n+1} - (1+\tau)^2\hat{\mathbf{W}}_i^n + \tau^2\hat{\mathbf{W}}_i^{n-1}}{\Delta^n t\,(1+\tau)} - \widehat{\mathrm{RHS}}_2\left(\{\hat{\mathbf{W}}_j^{n+1}\}_{j\in\overline{\mathcal{B}}_i}\right) & = & 0 \end{array}\right. \tag{51}$$

where $\tau = \frac{\Delta^n t}{\Delta^{n-1} t}$. Similarly to the first-order case, a linearization of $\mathrm{RHS}_2\left(\{\mathbf{W}_j^{n+1}\}_{j\in\overline{\mathcal{B}}_i}\right)$ and $\widehat{\mathrm{RHS}}_2\left(\{\hat{\mathbf{W}}_j^{n+1}\}_{j\in\overline{\mathcal{B}}_i}\right)$ must be carried out in order to avoid the solution of a nonlinear system at each time step. Clearly, the same approach as for the first order scheme could be considered, that is to approximate $\mathrm{RHS}_2\left(\{\mathbf{W}_j^{n+1}\}_{j\in\overline{\mathcal{B}}_i}\right)$ in the form:

$$\left\{\begin{array}{lcl} \mathrm{RHS}_2\left(\{\mathbf{W}_j^{n+1}\}_{j\in\overline{\mathcal{B}}_i}\right) & \simeq & \mathrm{RHS}_2\left(\{\mathbf{W}_j^{n}\}_{j\in\overline{\mathcal{B}}_i}\right) + \displaystyle\sum_{j\in\overline{\mathcal{B}}_i} \mathcal{D}_{2,ij}\Delta^n\mathbf{W}_j \\ \widehat{\mathrm{RHS}}_2\left(\{\hat{\mathbf{W}}_j^{n+1}\}_{j\in\overline{\mathcal{B}}_i}\right) & \simeq & \widehat{\mathrm{RHS}}_2\left(\{\hat{\mathbf{W}}_j^{n}\}_{j\in\overline{\mathcal{B}}_i}\right) + \displaystyle\sum_{j\in\overline{\mathcal{B}}_i} \hat{\mathcal{D}}_{2,ij}\Delta^n\hat{\mathbf{W}}_j \end{array}\right. \tag{52}$$

However, the linearization for the second-order accurate fluxes and the solution of the resulting linear system implies significant computational costs and memory requirements. This is a consequence of the more complex expression of second-order schemes with respect to their first-order counterparts (compare (23) with (26) and (32) with (41)) and, in particular, of the larger stencil of the second order flux function (i. e. considering an uniform triangular grid the set $\mathcal{B}_i$ contains 7 nodes while 19 nodes are in $\overline{\mathcal{B}}_i$). In order to reduce the computational costs, an alternative approach, considered in this work, is to use a defect-correction technique [14]. This method consists in iteratively solving simpler problems obtained by simply considering the same linearization used for the first-order scheme. The defect-correction iterations write as:

$$\left\{\begin{array}{l} \mathcal{W}^0 = \mathbf{W}^n \\ \mathcal{B}_i^s\Delta^s\mathcal{W}_i - \displaystyle\sum_{j\in N(i)} \mathcal{D}_{ij}^s\Delta^s\mathcal{W}_j = \mathcal{C}_i^s \quad s = 0,\cdots,r-1 \\ \mathbf{W}^{n+1} = \mathcal{W}^r \end{array}\right. \tag{53}$$

in which:

$$\begin{cases} \mathcal{B}_i^s = \dfrac{(1+2\tau)}{\Delta^n t\,(1+\tau)}\, I - \mathcal{D}_{ii}^s \\ \mathcal{D}_{ij}^s = \mathcal{D}_{ij}\left(\{\mathcal{W}_j^s\}_{j\in\mathcal{B}_i}\right) \\ \mathcal{C}_i^s = -\left(\dfrac{(1+2\tau)\mathcal{W}_i^s - (1+\tau)^2\mathbf{W}_i^n + \tau^2\mathbf{W}_i^{n-1}}{\Delta^n t\,(1+\tau)}\right) + \mathrm{RHS}_2\left(\{\mathcal{W}_j^s\}_{j\in\overline{\mathcal{B}}_i}\right) \end{cases} \tag{54}$$

$\mathcal{D}_{ij}$ being the generic matrices of the approximation (43); r is typically chosen equal to 2. Indeed, it can be shown [14, 15] that only one defect-correction iteration is theoretically needed to reach a second-order accuracy, while few additional iterations (one or two) can improve the robustness. An expression similar to (53)-(54), omitted for the sake of brevity, can be derived for the Modified Roe scheme by simply considering $\hat{\mathbf{W}}$ and $\widehat{\mathrm{RHS}}_2\left(\{\hat{\mathbf{W}}_j\}_{j\in\overline{\mathcal{B}}_i}\right)$ instead of $\mathbf{W}$ and $\mathrm{RHS}_2\left(\{\mathbf{W}_j\}_{j\in\overline{\mathcal{B}}_i}\right)$

4 Numerical Applications

The methodology proposed in section 3.4 to develop an implicit scheme starting from its explicit counterpart has been successfully applied both to the SRNH scheme and to the MR one. In this section two different numerical test-cases are considered in order to compare the different numerical approaches, explicit or implicit, as well as 1^{st} or 2^{nd}-order accurate, described previously. First, a 1D test-case is considered: even if in section 3 only the 2D formulations of the numerical schemes are outlined, the 1D formulation of the different numerical schemes can be easily recovered by simply setting to zero the velocity component along the y-axis and all the derivatives in the y direction. Then, a 2D test case is considered in order to show the main features of the implicit schemes with respect to the explicit ones.

4.1 1D Numerical Experiments

The proposed test-case is a standard numerical problem already considered in several papers (see e.g. [3, 6]). It is a sediment transport problem in a channel of length $l = 1000m$ with a non constant bottom profile.
The initial bottom topography is given by a hump shape function, that is:

$$\begin{cases} Z(0,x) &= \begin{cases} 0.1 + \sin^2\left(\dfrac{(x-300)\pi}{200}\right) & \text{if } 300 \le x \le 500 \\ 0.1 & \text{elsewhere} \end{cases} \\ h(0,x) &= 10 - Z(0,x) \\ u(0,x) &= \dfrac{10}{h(0,x)} \end{cases} \tag{55}$$

in which all the variables are in SI units. Two different uniform grids are considered for the discretization of the computational domain: a coarse grid, GR1, which is composed by 100 cells and a refined one, GR2, composed by 250 cells. The results computed by first and second order schemes, both explicit and implicit, are compared in terms of accuracy and computational costs. Four sets of simulations have been carried out, characterised by A_g set to $0.001, 0.01, 0.1$ and 1 respectively, A_g being the free parameter in the Grass model (4). Each value of the parameter A_g correspond to a specific speed of interaction between the flow and the bed-load and, as a consequence, to a specific time scale for the evolution of the bottom topography. The first value corresponds to a slow interaction between the flow and the bed-load, the last to a fast one, while the other values to intermediate conditions. Therefore, in order to observe significant variations of the bed profile, the simulations corresponding to small values of A_g are advanced in time for longer periods, as shown in table 1. All

A_g	1	0.1	0.01	0.001
Simulation time	700	7000	50000	500000

Table 1: Final simulation time (seconds) for the considered values of A_g.

the results and CPU times shown in the following are at the final instant of each simulation. All the simulations have been carried out on a 3 GHz Intel Pentium 4 processor with 2Gb RAM.

4.1.1 Slow speed of interaction between the bed-load and water flow

Figure 2 shows a comparison of the results obtained by means of the explicit version of the SRNH scheme at CFL= 0.8 with those of the implicit one at CFL= 1000, both for 1^{st} and 2^{nd}-order accuracy. Figure 3 shows the same comparison for the MR scheme. In both cases, there is practically no difference between the solutions obtained with the implicit and explicit version of the schemes, while the results obtained at 1^{st}-order of accuracy significantly differ from the 2^{nd}-order ones. Also note that the results shown in figure 2 and 3 for the second order implicit scheme are obtained using only one iteration of the Defect Correction method. Note also the similarity between the solutions computed by the SRNH and the MR schemes. As for the efficiency, both the SRNH and the MR implicit scheme seem to be unconditionally stable: the CFL has been increased up to 10^5 while obtaining stable solutions. However, the accuracy of the results obviously decreases if the time step is too large. As it is shown in figures 4 and 5, in the case of slow interaction between bed-load and water flow the quality of the results computed by the implicit schemes is not significantly deteriorated up to a CFL number of 1000. Increasing the CFL above 1000 significantly reduces the accuracy of the implicit schemes and, in the specific case of the second order scheme, can introduce unphysical oscillations, as shown in figures 4 and 5. These oscillations can be eliminated or at least reduced increasing the number of Defect Correction iterations, as shown

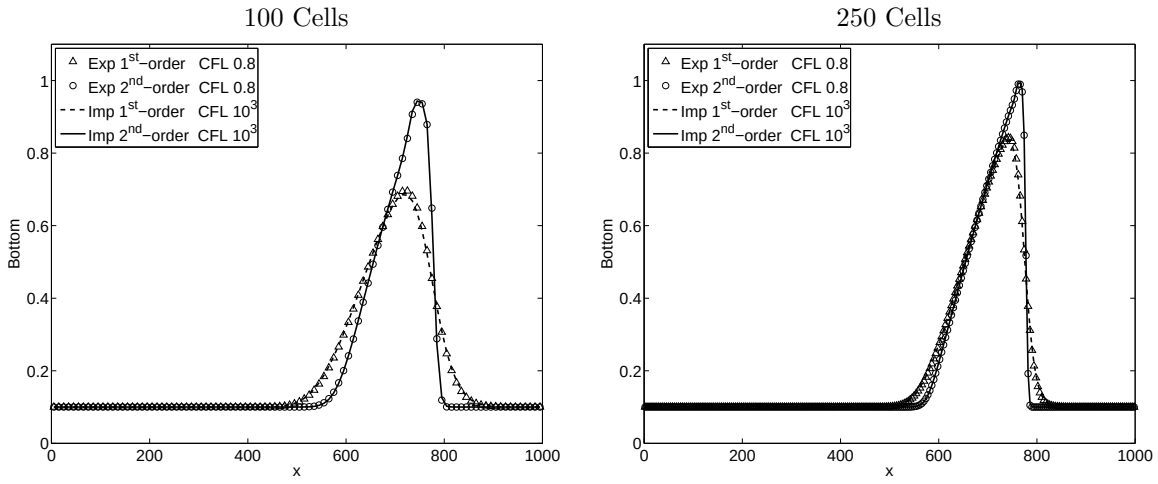

Figure 2: Bottom height results computed by the SRNH scheme and $A_g = 10^{-3}$: comparison between explicit and implicit, 1^{st} and 2^{nd}-order formulations.

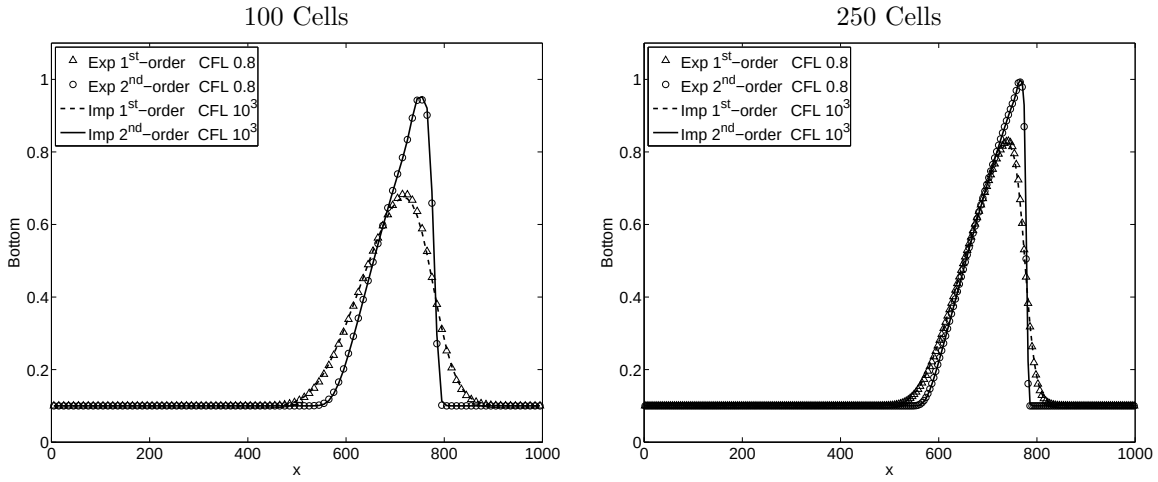

Figure 3: Bottom height results computed by the MR scheme and $A_g = 10^{-3}$: comparison between explicit and implicit, 1^{st} and 2^{nd}-order formulations.

in figure 6. Two Defect Correction iterations not only eliminate the oscillations but they also increase the accuracy of the numerical method and an additional iteration can lead to a slight improvement. For all the simulations considered in this work, no additional accuracy improved was observed by considering a number of Defect Correction larger than 3. For both the SRNH scheme and the MR one, the results computed using 3 Defect Correction iterations and CFL=10^4 are equivalent, in terms of accuracy, to those compute using 1 DeC

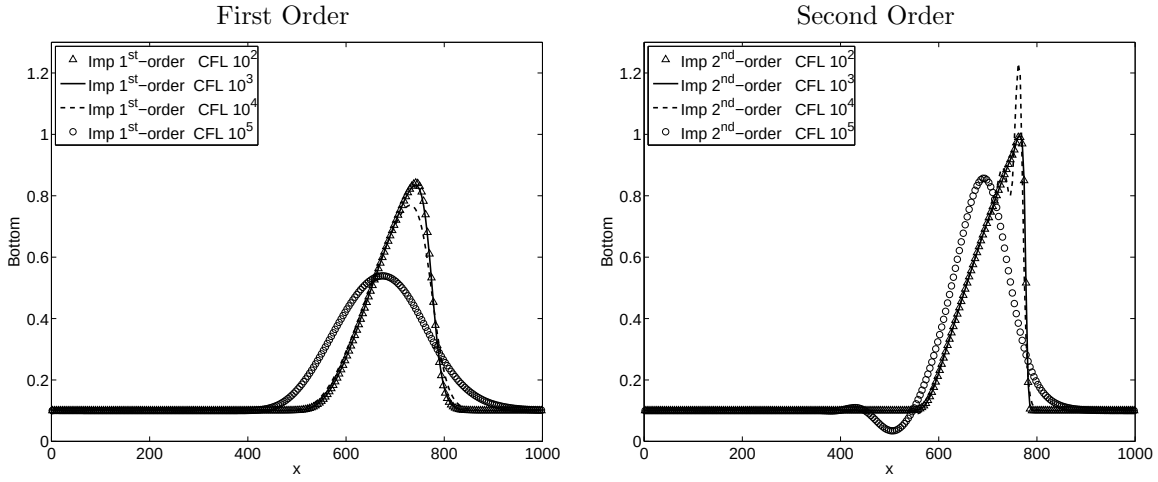

Figure 4: Bottom height obtained with the SRNH implicit scheme at different values of CFL, $A_g = 10^{-3}$, 250 Cells.

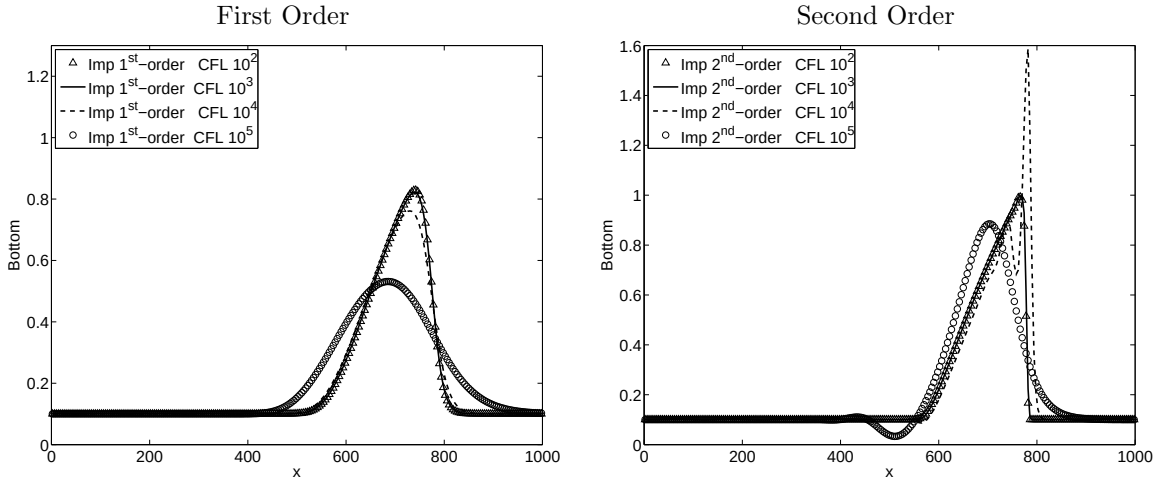

Figure 5: Bottom height obtained with the MR implicit scheme at different values of CFL, $A_g = 10^{-3}$, 250 Cells.

iteration at CFL= 10^3. As for the computational costs, table 2 show that already at CFL= 1000 the gain in CPU time obtained with the implicit scheme is large, for both the SRNH and MR schemes at both 1^{st} and 2^{nd}-order of accuracy. The CPU gain obtained with the implicit scheme is significantly larger for 2^{nd}-order accuracy. Indeed, when the implicit formulation is used, there are not significant differences, in terms of CPU time, between the 1^{st} and 2^{nd}-order simulations. Instead in the explicit case an important computational

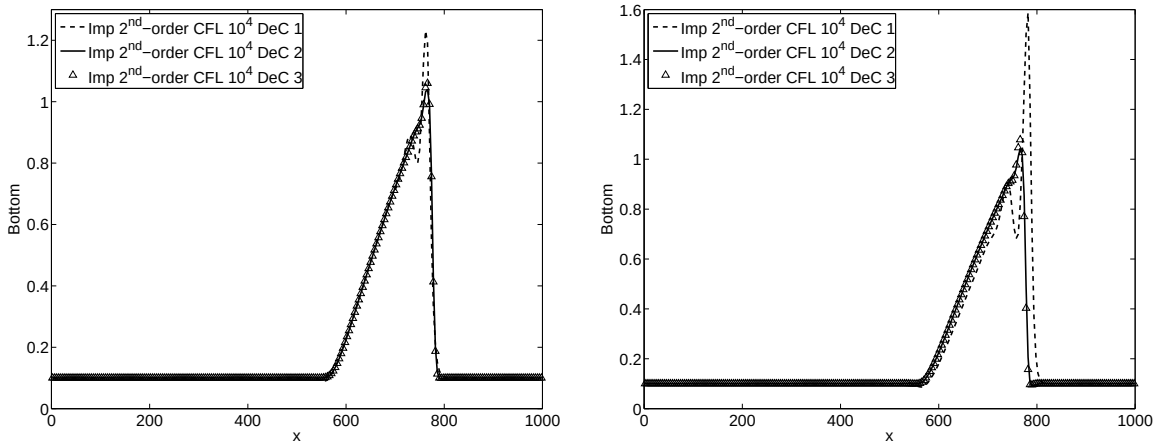

Figure 6: Effect of the number of Defect Correction iterations on the refined grid using a CFL value of 10^4: $A_g = 10^{-3}$; (left) SRNH scheme, (right) MR scheme.

	MR scheme		SRNH scheme	
Method	GR1	GR2	GR1	GR2
Explicit 1^{st} order, CFL= 0.8	$57.0s$	$355.6s$	$165.6s$	$1025.2s$
Explicit 2^{nd} order, CFL= 0.8	$136.7s$	$852.2s$	$247.2s$	$1537.1s$
Implicit 1^{st} order, CFL= 10^3	$0.3s$	$2.0s$	$0.4s$	$2.6s$
Implicit 2^{nd} order, CFL= 10^3, 1 DeC	$0.3s$	$2.1s$	$0.5s$	$3.0s$
Implicit 2^{nd} order, CFL= 10^4, 3 DeC	$0.1s$	$0.6s$	$0.1s$	$0.8s$

Table 2: CPU time required (seconds), case $A_g = 10^{-3}$.

cost increase is observed to reach 2^{nd}-order accuracy: the second order approach is $\simeq 1.5$ times more expensive than the first order approach for the SRNH scheme and $\simeq 2.4$ times for the MR one. As a consequence, already at CFL= 1000 using 1 DeC iteration the 2^{nd}-order implicit approach is more than 400 hundred times faster than the explicit one, for both the MR and SRNH schemes. The CPU gain of the 2^{nd}-order implicit approach can be further increased considering 3 DeC iterations and CFL= 10^4. However using more than one DeC iteration increases the memory requirement of the numerical method and the complexity of the algorithm. In the case of slow speed of interaction the gain in CPU time using 1 DeC iteration is remarkable so the use of 3 DeC iterations is probably not necessary. However in the following it will be shown that there are conditions in with using 3 DeC iterations can be useful.

4.1.2 Slow/Intermediate speed of interaction between bed-load and water flow

The increase of the A_g value allows to consider problems with stronger interaction between the bed-load and the water flow. A first intermediate case has been performed taking $A_g = 10^{-2}$. Also in this case, the implicit schemes seem to be unconditionally stable: the CFL has been increased up to 10^4 (only 6 iterations using the coarse grid to complete the tests and the simulations remained stable). Nevertheless, as shown in in figures 7 and 8, the upper bound for the CFL number of the 1^{st} and 2^{nd}-order implicit approaches is reduced of one order of magnitude for both the SRNH scheme and the MR one. Figures

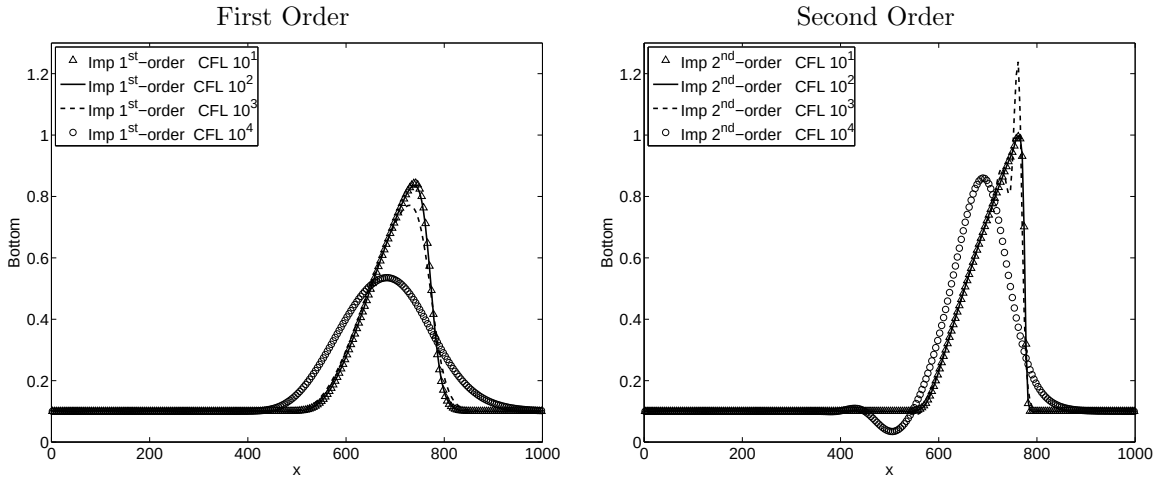

Figure 7: Bottom height obtained with the SRNH implicit scheme at different values of CFL, $A_g = 10^{-2}$, 250 Cells.

9 and 10 show a comparison of the results obtained by means of the explicit versions of the SRNH and MR schemes at CFL= 0.8 with those of the respective implicit versions at CFL= 100, both for 1^{st} and 2^{nd}-order accuracy for the case $A_g = 10^{-2}$. Considerations similar to the slow interaction case are still valid for this intermediate speed of interaction: there is practically no difference between the solutions obtained with the implicit and explicit schemes, while the results obtained at 1^{st}-order of accuracy significantly differ from the 2^{nd}-order ones. Similarly to the case $A_g = 10^{-3}$, an increase the CFL number above those values can induce unphysical oscillations in the solutions computed by the second order implicit scheme. Those oscillations can be reduced with additional Defect Correction iterations thus increasing the accuracy. As in the slow interaction case, the results computed using 1 Defect Correction iteration and CFL= 100 are equivalent to the result using CFL= 1000 and 3 DeC iterations. As for computational costs, table 3 show that even considering an intermediate speed of interaction between the bed-load and the water flow, with a lower upper bound

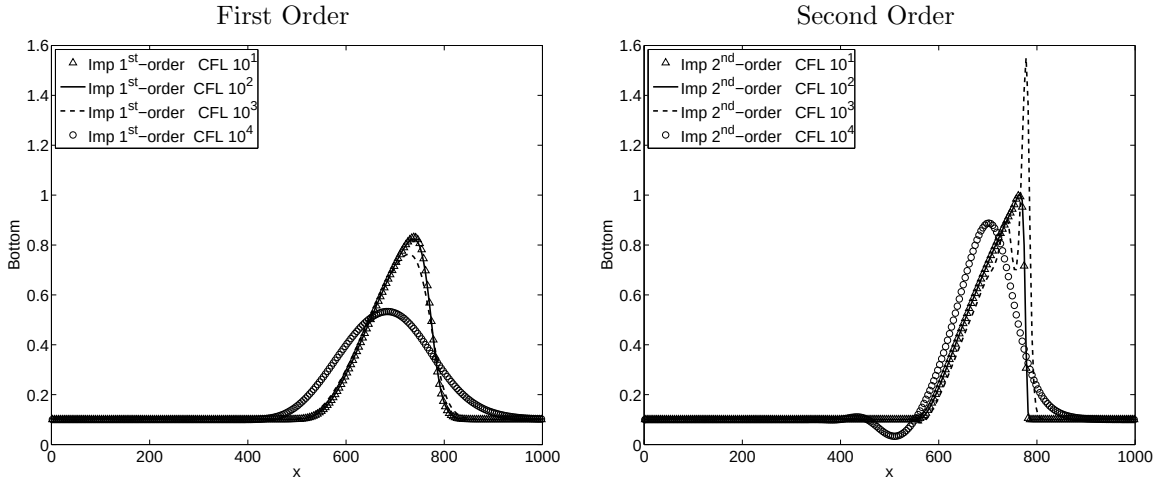

Figure 8: Bottom height obtained with the MR implicit scheme at different values of CFL, $A_g = 10^{-2}$, 250 Cells.

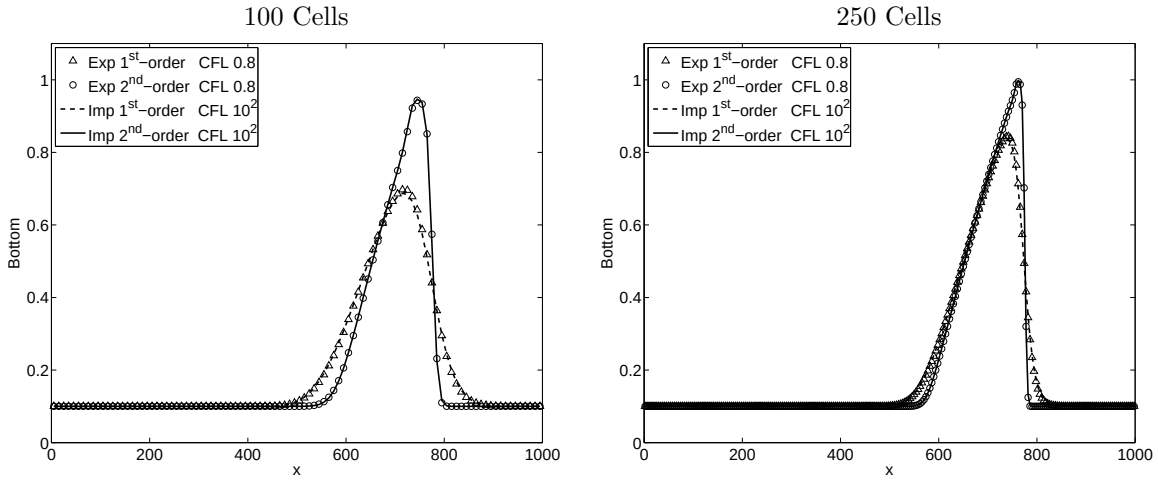

Figure 9: Bottom height results computed by the SRNH scheme and $A_g = 10^{-2}$: comparison between explicit and implicit, 1^{st} and 2^{nd}-order formulations.

on the CFL number, there is a gain in CPU time obtained with the implicit scheme, both at 1^{st} and 2^{nd}-order of accuracy. Even if the CPU time gain is not as large as the one in the previous case, the 2^{nd}-order implicit approach is still more than 40 times faster than the explicit one.

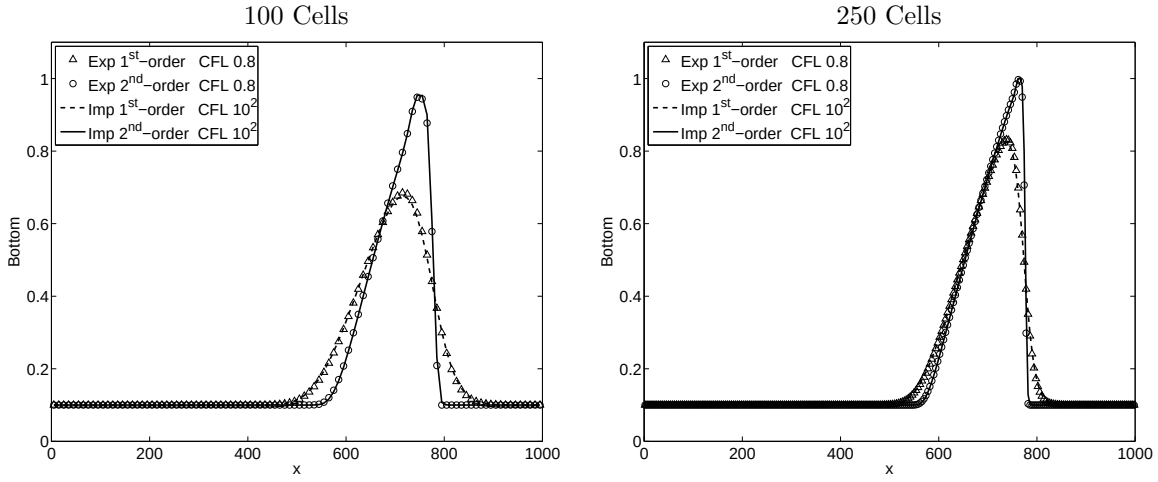

Figure 10: Bottom height results computed by the MR scheme and $A_g = 10^{-2}$: comparison between explicit and implicit, 1^{st} and 2^{nd}-order formulations.

	MR scheme		SRNH scheme	
Method	GR1	GR2	GR1	GR2
Explicit 1^{st} order, CFL= 0.8	$5.8s$	$35.7s$	$16.7s$	$103.3s$
Explicit 2^{nd} order, CFL= 0.8	$13.6s$	$85.6s$	$24.8s$	$153.7s$
Implicit 1^{st} order, CFL= 10^2	$0.3s$	$2.0s$	$0.4s$	$2.6s$
Implicit 2^{nd} order, CFL= 10^2, 1 DeC	$0.3s$	$2.1s$	$0.5s$	$3.0s$
Implicit 2^{nd} order, CFL= 10^3, 3 DeC	$0.1s$	$0.6s$	$0.1s$	$0.8s$

Table 3: CPU time required (seconds), case $A_g = 10^{-2}$.

4.1.3 Intermediate/Fast speed of interaction between bed-load and water flow

This case correspond to $A_g = 10^{-1}$. The conclusions drawn for the previous case hold also for this case: the performance of the implicit scheme are similar for the SRNH and the MR schemes. The implicit schemes at CFL = 10 and the explicit scheme at CFL=0.8 give similar results, as shown in figures 11 and 12, and the implicit schemes seem to be unconditionally stable. However, in order to prevent loss in accuracy, in this case a further reduction of the maximum time step must be considered. Figures similar to 7 and 8 (leaved out for the sake of brevity) can show that the upper limit for the CFL number must be set to 10 in this case. Notwithstanding this lower limitation, the implicit scheme, both for SRNH and the MR scheme, is still preferable in terms of CPU time, even if the gain is reduced with respect to the previous cases, as shown in table 4. In this case the 2^{nd}-order implicit approach with 1 DeC iteration is roughly 4 times faster than its explicit counterpart. As a consequence, for this value of

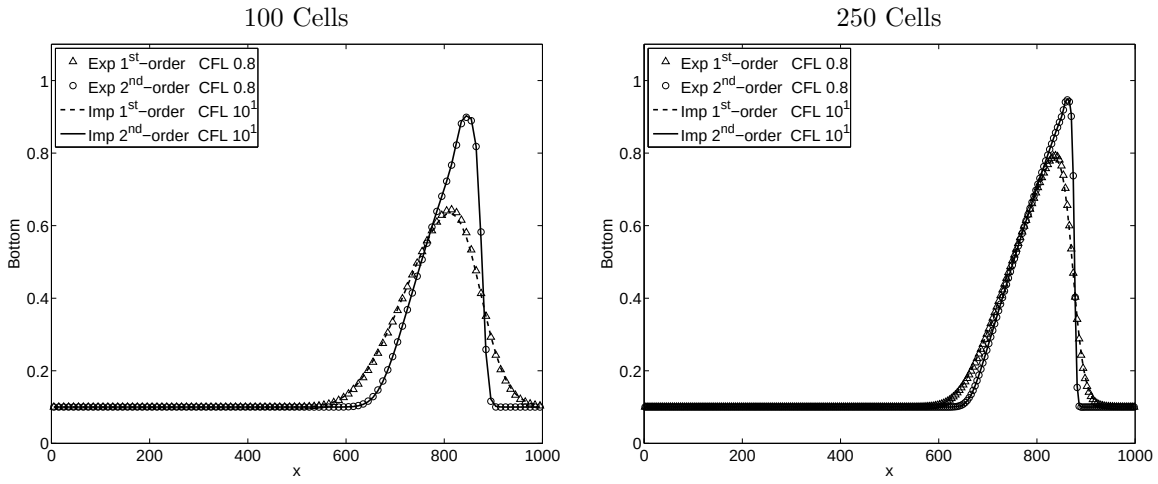

Figure 11: Bottom height results computed by the SRNH scheme and $A_g = 10^{-1}$: comparison between explicit and implicit, 1^{st} and 2^{nd}-order formulations.

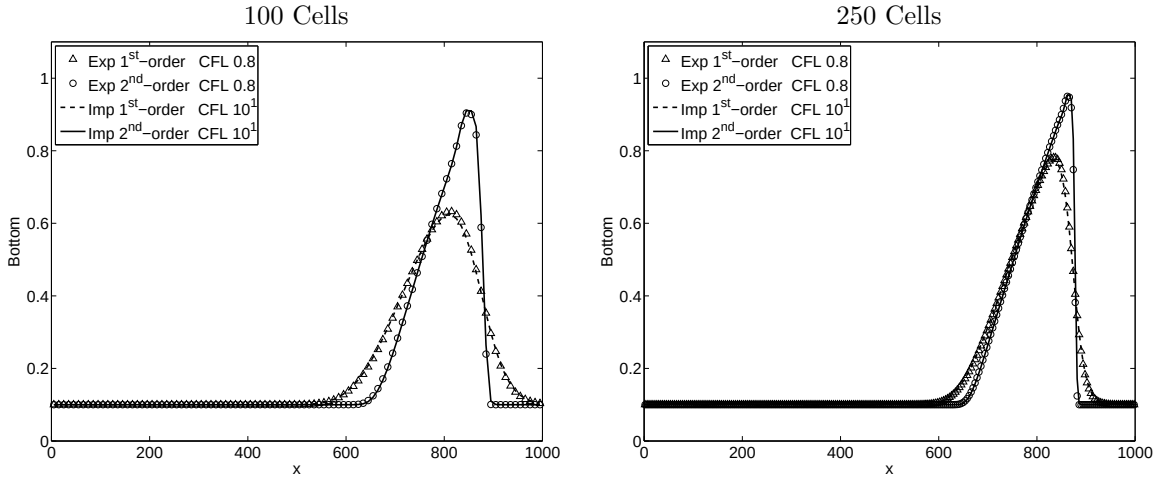

Figure 12: Bottom height results computed by the MR scheme and $A_g = 10^{-1}$: comparison between explicit and implicit, 1^{st} and 2^{nd}-order formulations.

A_g it could be interesting to use 3 DeC iterations with a consequent CPU time gain of $\simeq 12$ times for both the MR and SRNH schemes.

4.1.4 Fast speed of interaction between bed-load and water flow

Figures 13 and 14 show a comparison of the results obtained by means of the explicit versions of the SRNH and MR schemes at CFL= 0.8 with those of the

	MR scheme		SRNH scheme	
Method	GR1	GR2	GR1	GR2
Explicit 1^{st} order, CFL= 0.8	$0.8s$	$5.1s$	$2.4s$	$14.6s$
Explicit 2^{nd} order, CFL= 0.8	$2.0s$	$12.1s$	$3.6s$	$21.8s$
Implicit 1^{st} order, CFL= 10^3	$0.5s$	$2.9s$	$0.6s$	$3.7s$
Implicit 2^{nd} order, CFL= 10^1, 1 DeC	$0.5s$	$3.0s$	$0.7s$	$4.2s$
Implicit 2^{nd} order, CFL= 10^2, 3 DeC	$0.1s$	$0.9s$	$0.2s$	$1.1s$

Table 4: CPU time required (seconds), case $A_g = 10^{-1}$.

implicit versions at CFL= 1, both for 1^{st} and 2^{nd}-order accuracy. As for the previous case there is no difference between the solutions obtained with the implicit and explicit schemes. In this case of fast interaction between bed-load

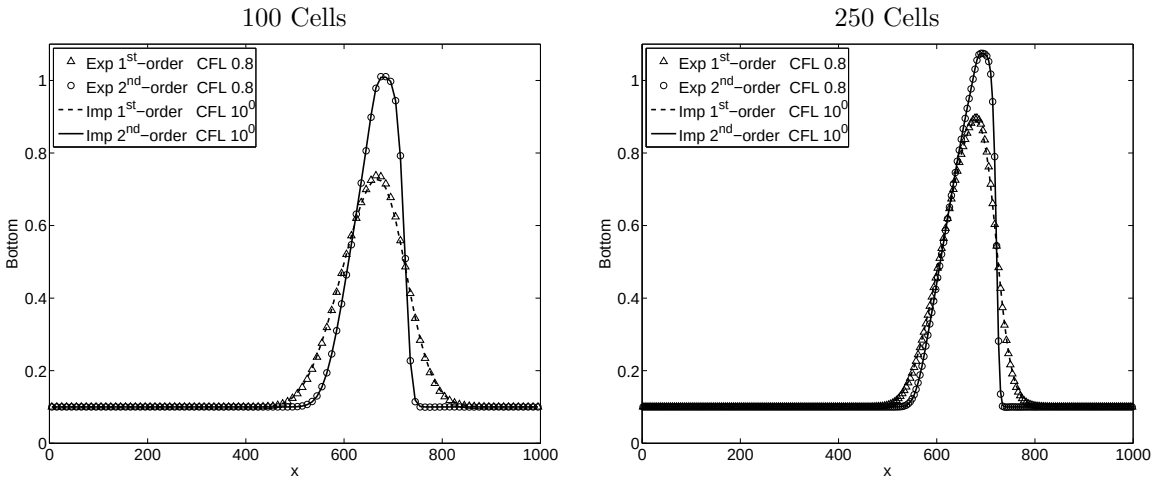

Figure 13: Bottom height results computed by the SRNH scheme and $A_g = 10^{-0}$: comparison between explicit and implicit, 1^{st} and 2^{nd}-order formulations.

and water flow, the quality of the results for the implicit schemes imposes a maximum CFL number equal to 1, although the implicit schemes seem again to be unconditionally stable. As a consequence, as is shown in table 5 in this test case the computational cost of the first order implicit approach is larger than for the explicit one. Also the 2^{nd} order implicit method with 1 DeC iteration is more expensive than its explicit counterpart. The only case in witch the implicit approach seems to be more efficient than the explicit scheme seems to be the second-order method with 3 DeC iterations. For this speed of interaction the use of multiple DeC iterations seems to be mandatory. However due the small time required, to really quantify the gain for this case, more demanding cases are required. Summarising, considering the 1D case, the implicit time advancing seems to be unconditionally stable in all the considered cases, the

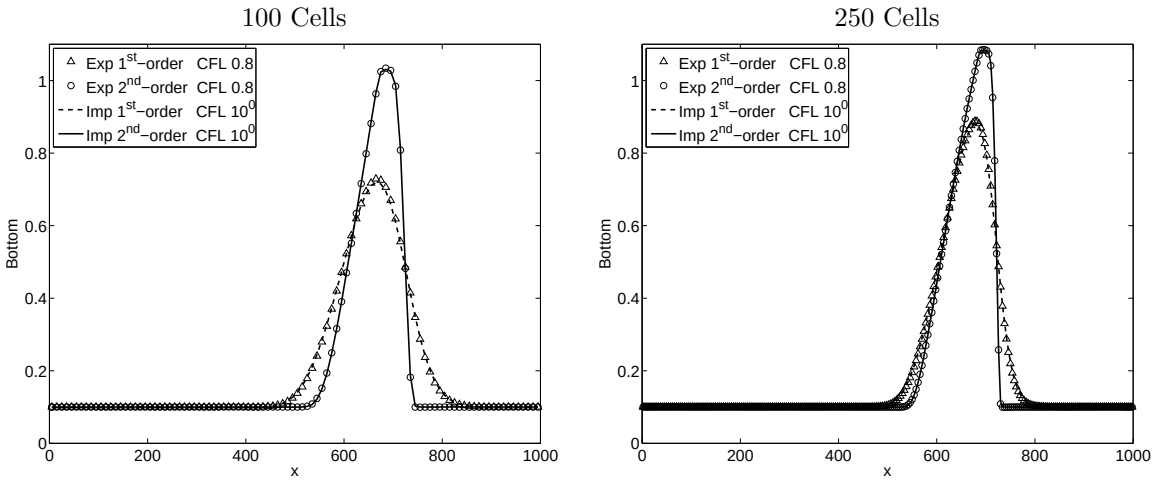

Figure 14: Bottom height results computed by the MR scheme and $A_g = 10^{-0}$: comparison between explicit and implicit, 1^{st} and 2^{nd}-order formulations.

	MR scheme		SRNH scheme	
Method	GR1	GR2	GR1	GR2
Explicit 1^{st} order, CFL= 0.8	$0.1s$	$0.6s$	$0.3s$	$1.8s$
Explicit 2^{nd} order, CFL= 0.8	$0.2s$	$1.4s$	$0.4s$	$2.6s$
Implicit 1^{st} order, CFL= 10^3	$0.6s$	$3.6s$	$0.7s$	$4.4s$
Implicit 2^{nd} order, CFL= 10^0, 1 DeC	$0.6s$	$3.6s$	$0.8s$	$5.2s$
Implicit 2^{nd} order, CFL= 10^1, 3 DeC	$0.2s$	$1.0s$	$0.2s$	$1.3s$

Table 5: CPU time required (seconds), case $A_g = 1$.

CFL limitation to avoid loss of accuracy is roughly inversely proportional to A_g and, in presence of unphysical oscillations, increasing the number of Defect Correction iterations can significantly increase the accuracy. The implicit time advancing is computationally efficient even using only one DeC iteration for slow and intermediate speeds of interaction. For fast speed of interaction 3 DeC iterations are required, as show in figure 15.

4.2 2D Numerical Experiments

The 2D test case considered herein is a well-known benchmark test, proposed in several papers [3, 8] and it is the 2D generalisation of the 1D test-case described in section 4.1. It is a sediment transport problem in a square domain Ω of dimensions $1000 \times 1000m^2$ with a non constant bottom relief.

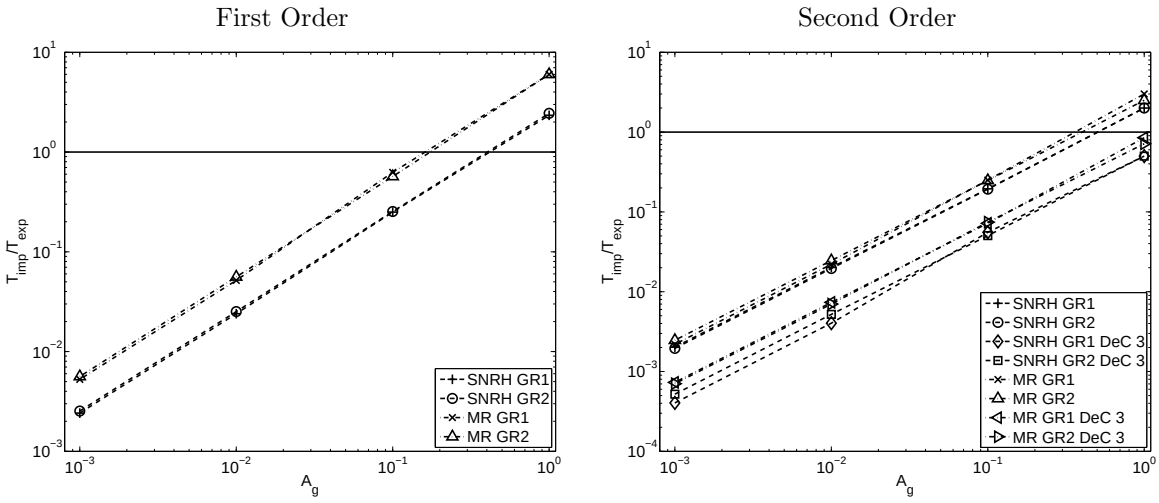

Figure 15: Ratio between the implicit computational time and the explicit one as a function of A_g.

The initial bottom topography is defined as follows:

$$\begin{cases} Z(0,x) & = & \begin{cases} \sin^2\left(\dfrac{(x-300)\pi}{200}\right)\sin^2\left(\dfrac{(y-400)\pi}{200}\right) & \text{if } (x,y)\in Q_h \\ 0 & & \text{elsewhere in } \Omega \end{cases} \\ h(0,x,y) & = & 10 - Z(0,x,y) \\ u(0,x,y) & = & \dfrac{10}{h(0,x,y)} \\ v(0,x,y) & = & 0 \end{cases} \tag{56}$$

where $Q_h = [300, 500] \times [400, 600]$. Dirichlet boundary conditions are imposed at the inlet, while at the outlet characteristic based conditions are used. Finally, free-slip is imposed on the lateral boundaries. The spatial discretization of the computational domain has been carried out by using two different grids. Both of them are symmetric with respect to the axis $y = 500m$ and are composed by uniform-size triangular elements. The characteristic length l_m of the elements is $l_m = 20$ and $l_m = 10$ for, respectively the coarse grid GR1 and the refined one, GR2. The main characteristics of the grids used are reported in table 6. For the 2D case two different values of the parameter A_g are considered, namely

	Nodes	Elements	l_m
GR1	2901	5600	20
GR2	11425	22448	10

Table 6: Main characteristics of the grids used in the simulations.

the slow interaction case, $A_g = 0.001$ and the fast one, $A_g = 1$. Considering

the results obtained in the 1D case, this working conditions corresponds to the most favourable condition for, respectively, the implicit time-advancing and the explicit one. Since different values of the parameter A_g corresponds to different time scales for the evolution of the bottom topography, different time intervals have been simulated for the two considered cases. In particular for the case $A_g = 1$ the total simulation time is 500 seconds while for the case $A_g = 0.001$ is 100 hours (360000 seconds). All the results and CPU times shown in the following are at the final instant of each simulation.

4.2.1 Fast interaction between the bed-load and water flow

For the fast speed of interaction case, in terms of accuracy the results essentially confirm the analysis of the 1D case. For this value of A_g, to avoid loss of accuracy the CFL number of the implicit scheme must be lowered down to 1. As an example figure 16 shows a comparison between the explicit and implicit approach at different CFL values for the MR scheme. On the other hand, by increasing the number of DeC iterations, it is possible to increase the maximum CFL value by a factor 10 for the second order implicit approach without losing in accuracy, as shown in figure 17. Considering the stability of the numerical method, we found that using CFL= 10^2 the SRNH scheme is unstable on the refined grid. It is possible that this stability limitation is the consequence of a non-optimal choice for the linear solver associated with the implicit approach. However, since using such an high CFL number, the quality of the results is highly decreased this possibility has not been investigated. A new feature deserving attention is the effect on the results of the solution of the linear system associated with the implicit scheme. In the 1D case the linear system was solved using a direct method thus, neglecting roundoff errors, computing the exact solution. Instead, for the 2D case, the linear system associated with the implicit scheme is solved using an iterative GMRES method. This iterative solver naturally introduces an additional approximation, since it is necessary to define a criterion in order to stop the iteration loop.

In this work the iterative solver is terminated when the residual of the linear system is below a fixed fraction of the initial residual, that is:

$$\text{res} \leq \text{res}_0 * Tol$$

Clearly, the smaller Tol, the smaller is the influence of the linear solver on the results. However, exceedingly decreasing Tol can abruptly increase the computational costs and quickly overcome any advantage coming from small values for Tol. Whatever is the CFL number, for the Modified Roe scheme we found that a reasonable value is $Tol = 10^{-4}$. The behaviour of the SRNH scheme is different from the MR one: for the first order SRNH setting $Tol = 10^{-6}$ is enough to preserve the accuracy of the method. Instead the second order scheme requires a different approach: using $Tol = 10^{-6}$ the solution computed by the implicit scheme is not completely similar to the explicit one. Lowering Tol below 10^{-6} could introduce roundoff errors, contaminating the solution. Under these circumstances, it is interesting to compare the effect of the value

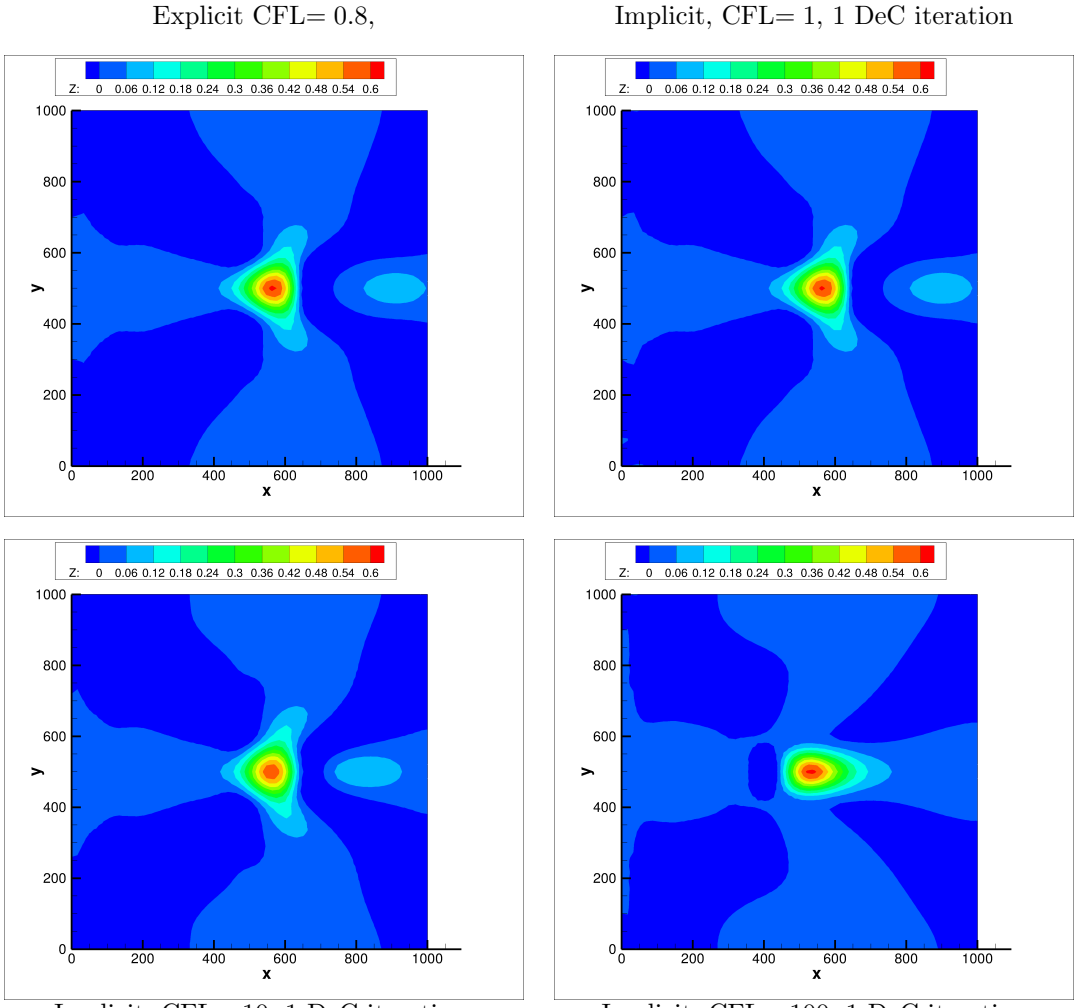

Figure 16: Comparison of the results for the bed profile of the 2^{nd}-order MR scheme, $A_g = 1$, Grid GR1.

Tol with respect the number of Defect Correction iterations. Figure 18 shows that increasing the number of DeC iterations have a more pronounced effect than decreasing *Tol*. The CPU times for all the considered schemes and grids are reported in 7. An interesting feature of the 2D case is that considering a fast interaction speed, the MR scheme at CFL= 10 with 3 DeC iterations is computationally more efficient than its explicit counterpart. This is not true for the SRNH scheme, as shown in table 7. The origin of this different behaviour is related to the different formulations of the two numerical methods: the second order extension of the MR scheme is more computationally demanding that

Explicit CFL= 0.8, Implicit, CFL= 10, 3 DeC iteration

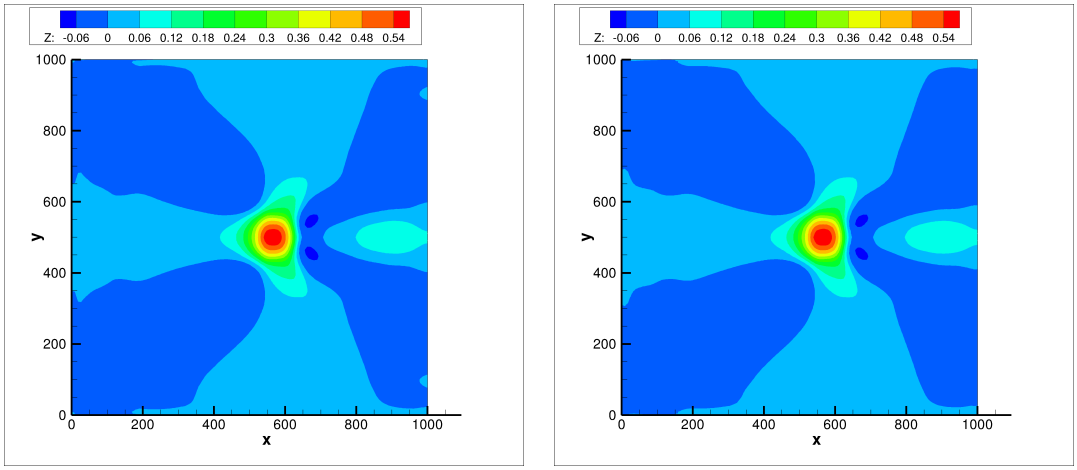

Figure 17: Comparison of the results for the bed profile of the 2^{nd}-order SRNH scheme, $A_g = 1$, Grid GR2.

the SRNH one (see equation (37)) and, as a consequence, a Defect Correction approach is more efficient in this case. Also, the linear system associated to the implicit SRNH scheme seem to be more demanding than the MR one and this increase the computational time.

	MR scheme		SRNH scheme	
Method	GR1	GR2	GR1	GR2
Explicit 1^{st} order, CFL= 0.8	16.1s	130.4s	21.0s	169.7s
Explicit 2^{nd} order, CFL= 0.8	140.4s	1137s	52.4s	409.9s
Implicit 1^{st} order, CFL= 1	197.0s	1560s	191.5s	1541s
Implicit 2^{nd} order, CFL= 1, 1 DeC	239.0s	1919s	198.7s	1582s
Implicit 2^{nd} order, CFL= 10, 3 DeC	84.3s	689.0s	74.5s	606.8s

Table 7: CPU time required (seconds), case $A_g = 1$.

4.2.2 Slow interaction between the bed-load and water flow

For the slow speed of interaction case, the results of the 2D computations essentially confirm the analysis of the 1D case. Both for the SRNH and for the MR scheme there is practically no difference between the solutions obtained with the implicit schemes at CFL= 1000 and explicit ones at CFL= 0.8, while the solutions computed using 1^{st}-order of accuracy significantly differ from the 2^{nd}-order ones as shown in figures 19 and 20. Again, by increasing the number of DeC iterations for the second order scheme, an increase of the CFL number limit without losing accuracy can be obtain, as show in figure 21. As for the fast

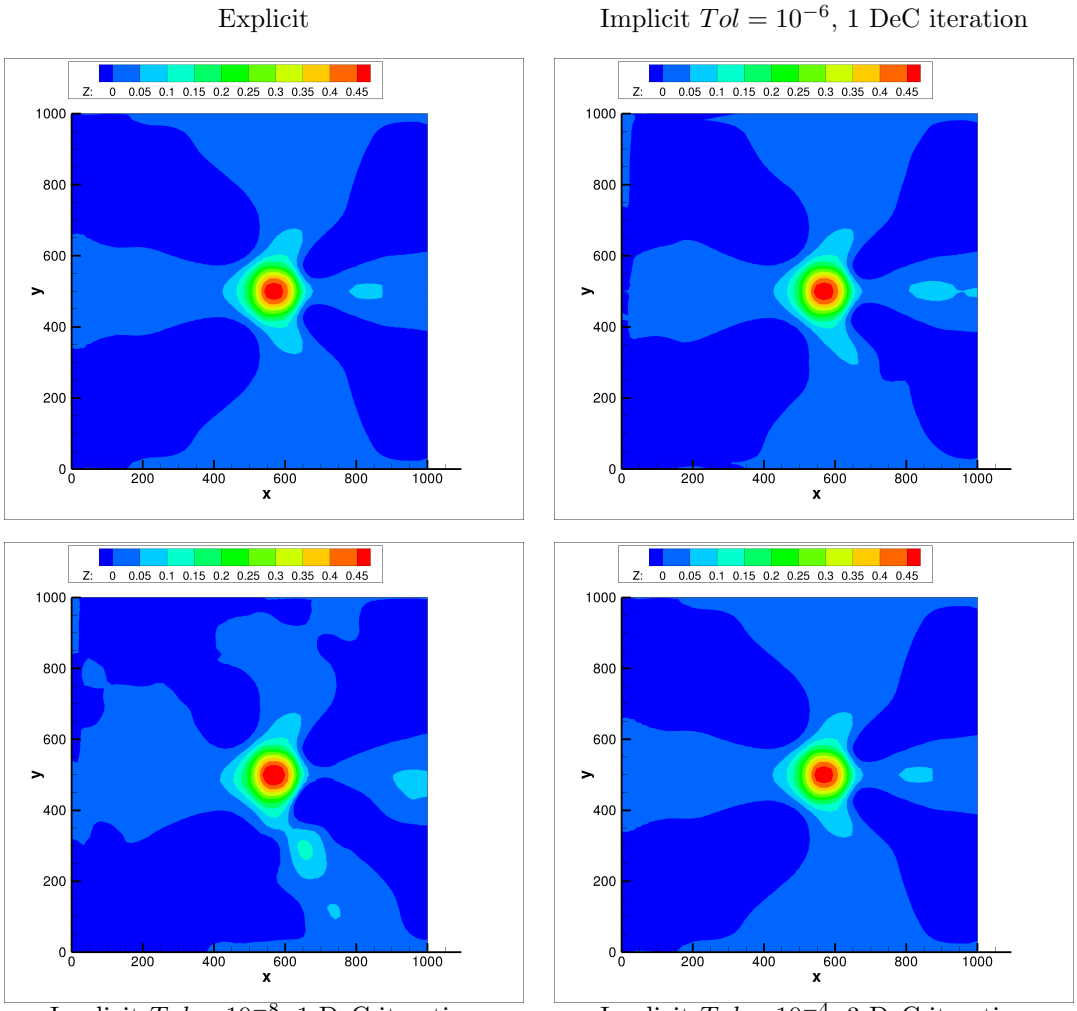

Figure 18: Comparison of the effect of the value of Tol and of the number of DeC iterations for the SRNH scheme on the bed profile, CFL= 1, GR1.

speed of interaction case, the MR implicit scheme seems to be unconditionally stable while the SRNH implicit scheme is unstable using a CFL number of 10^5 using the grid GR2. Finally, as for the computational costs, for both the SRNH and MR schemes, already at CFL= 1000 the gain in CPU time obtained with the implicit scheme is large, both at 1^{st} and 2^{nd}-order of accuracy as shown in table 8. Increasing the number of DeC iteration together with the CFL number can further decrease the computational cost of the second order implicit scheme.

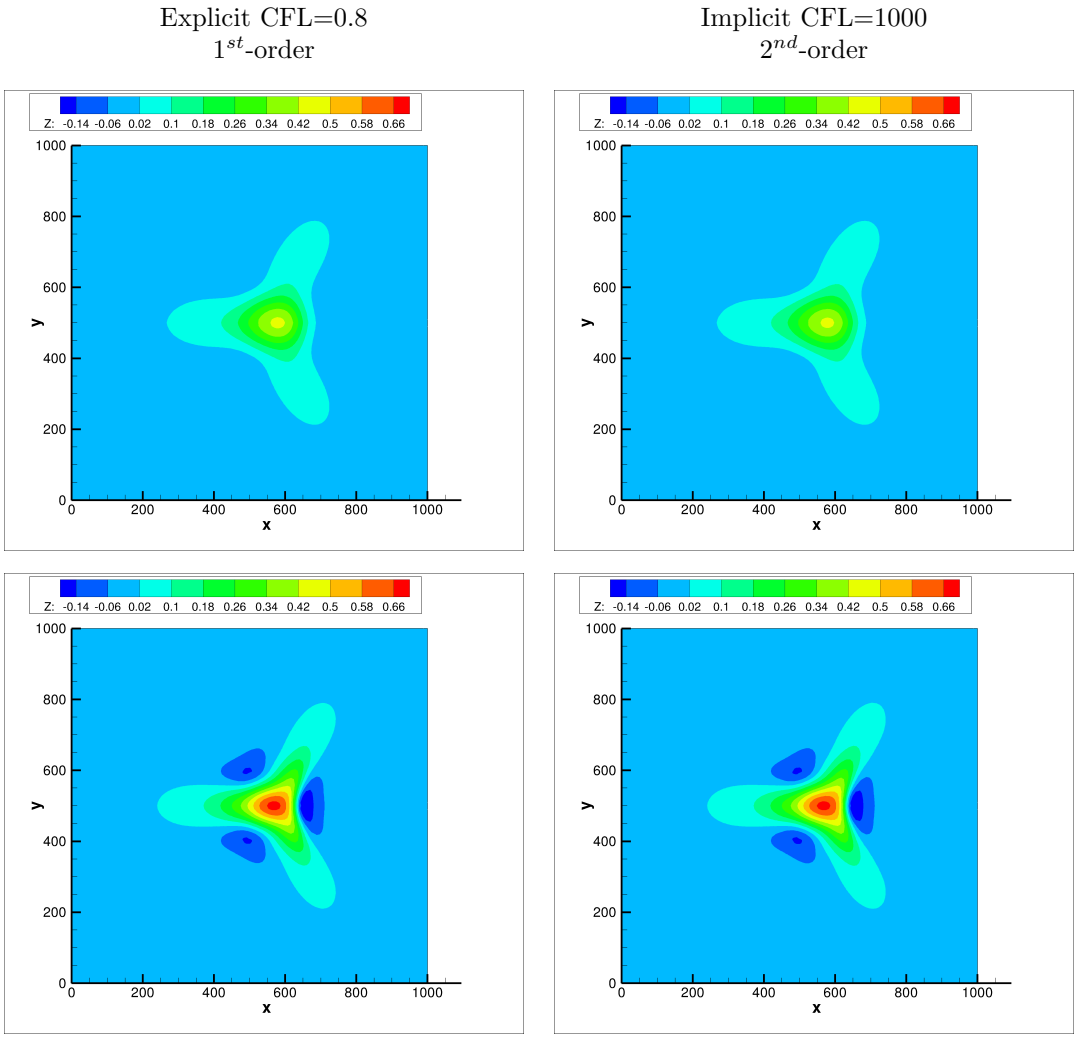

Figure 19: Comparison of the results of the bed profile of the explicit and implicit MR scheme schemes, for 1^{st} (top) and 2^{nd}-order (bottom) of accuracy: $A_g = 10^{-3}$,Grid GR2.

5 Concluding remarks

The focus of the present paper was on the comparison between implicit and explicit schemes for the simulation of sediment transport problems, in terms of accuracy and computational requirements The implicit schemes have been generated starting from their explicit counterpart using an automatic differentiation tool, Tapenade. In the contest of the Shallow-Water model 1D and 2D test cases have been considered, characterised by different rates of interaction

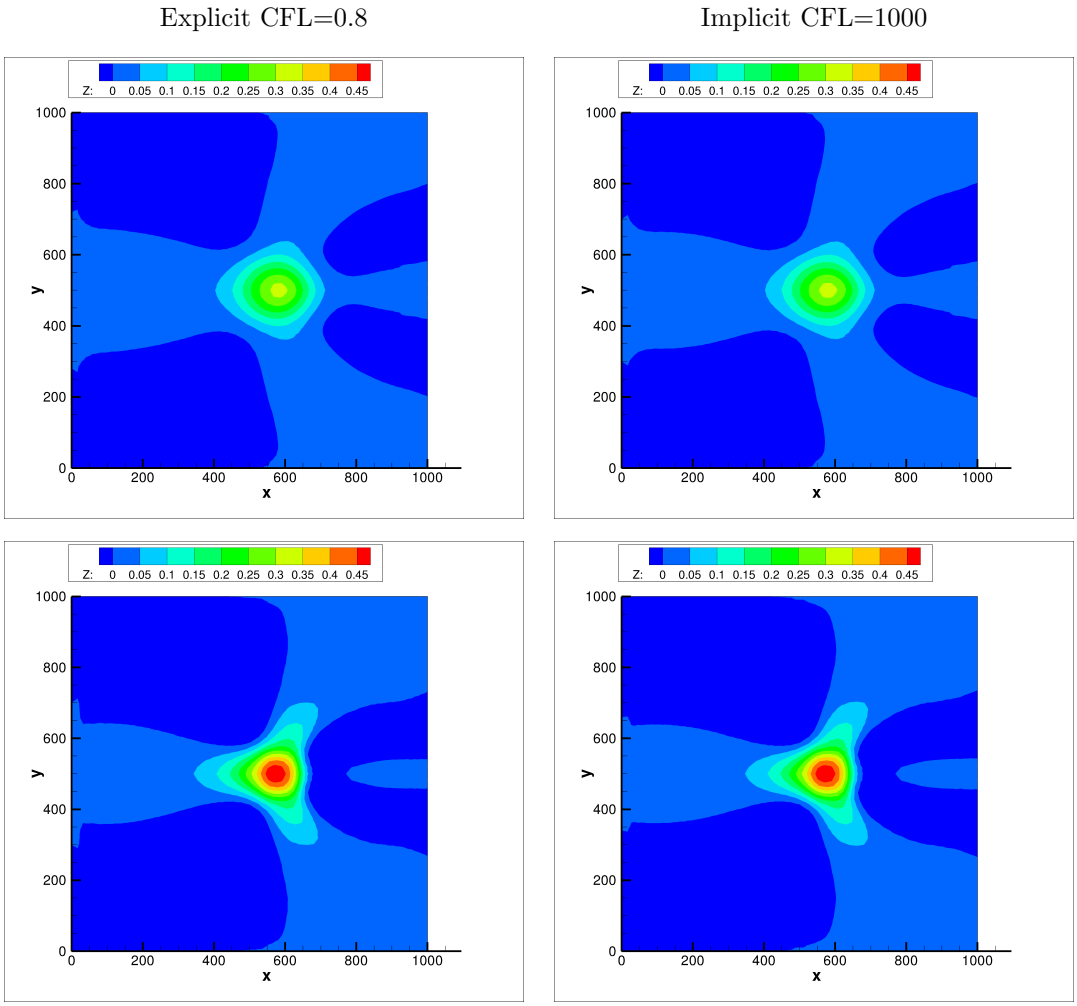

Figure 20: Comparison of the results of the bed profile of the explicit and implicit SRNH scheme schemes, for 1^{st} (top) and 2^{nd}-order (bottom) of accuracy: $A_g = 10^{-3}$,Grid GR1.

between the bed and the water flow. In the 1D simulations, for both the SRNH and MR schems, the implicit method was found to run with a virtually unlimited CFL number without stability problems. However, to avoid loss of accuracy, the CFL number of the implicit scheme must be reduced to a value roughly inversely proportional to the constant determining the velocity of the interaction between the flow and the bed-load. Only one DeC iteration seem to be enough to preserve the accuracy of the second order implicit scheme even if increasing the number of DeC iteration can increase the maximum CFL number allowable. In the 1D case the implicit code has been found to be computationally

CFL= 10^3 with 1 DeC iteration, CFL= 10^4 with 3 DeC iteration

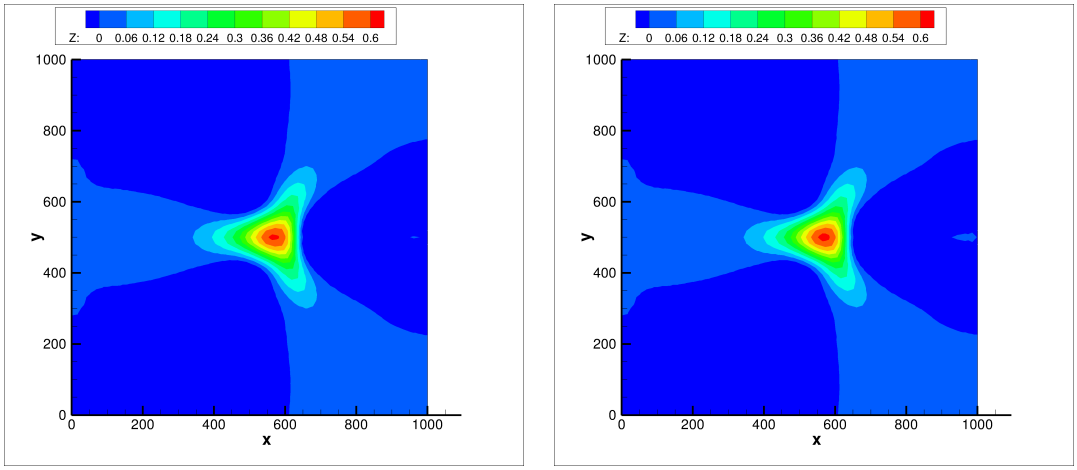

Figure 21: Comparison of the results for the bed profile of the 2^{nd}-order implicit MR scheme, $A_g = 10^{-3}$, Grid GR1.

	MR scheme		SRNH scheme	
Method	GR1	GR2	GR1	GR2
Explicit 1^{st} order, CFL= 0.8	$9497s$	$78073s$	$12824s$	$103238s$
Explicit 2^{nd} order, CFL= 0.8	$82993s$	$670770s$	$30996s$	$247215s$
Implicit 1^{st} order, CFL= 10^3	$199.8s$	$2134s$	$323.6s$	$4336s$
Implicit 2^{nd} order, CFL= 10^3, 1 DeC	$293.7s$	$2776s$	$481.5s$	$8537s$
Implicit 2^{nd} order, CFL= 10^4, 3 DeC	$136.4s$	$1625s$	$265.9s$	$4866s$

Table 8: CPU time required (seconds), case $A_g = 10^{-3}$.

more efficient than the explicit one for all the considered rates of interactions between the bed and the flow. The 2D tests globally confirm these results, but the influence of the number of Defect Correction iterations seems to be more pronounced, in particular for the SRNH scheme. Furthermore, it was found that for fast bedload/water interaction only the MR the implicit scheme can be competitive with its explicit counterpart. As a consequence the proposed methodology, implicit time advancing and Defect Correction technique, seems to be particularly suitable for slow and intermediate speed of interaction and for computationally expensive explicit methods. Also, since the CFL number is limited by the speed of interaction, another field deserving attention is the application of the considered methodology to more complex models for the evolution of the bed.

A Additional details on the SRNH Scheme

In this appendix, the explicit expressions of some quantities appearing in section 3.2 are given. Given two generic states U_L and U_R, the function $\overline{\mathbf{U}}\left(\mathbf{U}_L, \mathbf{U}_R\right)$ which returns a Roe averaged state is computed as:

$$\left\{\begin{array}{lcl} \overline{\mathbf{U}}\left(\mathbf{U}_L, \mathbf{U}_R\right) & = & \left(\overline{h}_{LR}, \quad \overline{u}_{\eta,LR}, \quad \overline{u}_{\tau,LR}, \quad \overline{Z}_{LR}\right)^T \\ \overline{h}_{LR} & = & \dfrac{h_L + h_R}{2} \\ \overline{u}_{\eta,LR} & = & \dfrac{u_{\eta,L}\sqrt{h_L} + u_{\eta,R}\sqrt{h_R}}{\sqrt{h_L} + \sqrt{h_R}} \\ \overline{u}_{\tau,LR} & = & \dfrac{u_{\tau,L}\sqrt{h_L} + u_{\tau,R}\sqrt{h_R}}{\sqrt{h_L} + \sqrt{h_R}} \\ \overline{Z}_{LR} & = & \dfrac{Z_L + Z_R}{2} \end{array}\right. \tag{57}$$

The definition of the sign matrix $\mathrm{sgn}\left[\mathbf{A}_\eta(\overline{\mathbf{U}})\right]$ appearing in (19) is the following:

$$\mathrm{sgn}\left[\mathbf{A}_\eta(\overline{\mathbf{U}})\right] = \mathcal{R}(\overline{\mathbf{U}})\mathrm{sgn}\left[\Lambda(\overline{\mathbf{U}})\right]\mathcal{R}^{-1}(\overline{\mathbf{U}}) \tag{58}$$

where $\Lambda(\overline{\mathbf{U}})$ is the diagonal matrix of eigenvalues of $\mathbf{A}_\eta(\overline{\mathbf{U}})$, $\mathcal{R}(\overline{\mathbf{U}})$ is the corresponding right-eigenvectors matrix and $\mathbf{A}_\eta(\mathbf{U})$ is defined as:

$$\mathbf{A}_\eta(\mathbf{U}) = \begin{pmatrix} u_\eta & h & 0 & 0 \\ g & u_\eta & 0 & g \\ 0 & 0 & u_\eta & 0 \\ 0 & A_g\xi(3u_\eta^2 + u_\tau^2) & 2A_g\xi u_\eta u_\tau & 0 \end{pmatrix} \tag{59}$$

B Additional details on the Modified Roe Scheme

In this appendix, the explicit expressions of some quantities appearing in section 3.3 are given. First, given two generic states $\hat{\mathbf{W}}_L$ and $\hat{\mathbf{W}}_R$, it is possible to define an additional intermediate Roe state which components in primitive variables are:

$$\left\{\begin{array}{lcl} \hat{h}_{LR} & = & \dfrac{\hat{h}_L + \hat{h}_R}{2} \\ \hat{u}_{LR} & = & \dfrac{\hat{u}_L\sqrt{\hat{h}_L} + \hat{u}_R\sqrt{\hat{h}_R}}{\sqrt{\hat{h}_L} + \sqrt{\hat{h}_R}} \\ \hat{v}_{LR} & = & \dfrac{\hat{v}_L\sqrt{\hat{h}_L} + \hat{v}_R\sqrt{\hat{h}_R}}{\sqrt{\hat{h}_L} + \sqrt{\hat{h}_R}} \end{array}\right. \tag{60}$$

Additional useful definitions related to the averaged Roe states are:

$$\begin{cases} \hat{c}_{LR} &= \sqrt{g\hat{h}_{LR}} \\ \hat{u}^2_{LR} &= \dfrac{\hat{u}^2_L\sqrt{\hat{h}_R}+\hat{u}^2_R\sqrt{\hat{h}_L}}{\sqrt{\hat{h}_L}+\sqrt{\hat{h}_R}} \\ \hat{v}^2_{LR} &= \dfrac{\hat{v}^2_L\sqrt{\hat{h}_R}+\hat{v}^2_R\sqrt{\hat{h}_L}}{\sqrt{\hat{h}_L}+\sqrt{\hat{h}_R}} \end{cases} \tag{61}$$

The function $\hat{\mathbf{B}}\left(\hat{\mathbf{W}}_L,\hat{\mathbf{W}}_R\right)$ appearing in (32) is defined as follows:

$$\hat{\mathbf{B}}\left(\hat{\mathbf{W}}_L,\hat{\mathbf{W}}_R\right)=\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & g\hat{h}_{LR}\mathbf{n}_{x,LR} \\ 0 & 0 & 0 & g\hat{h}_{LR}\mathbf{n}_{y,LR} \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{62}$$

Finally, the functions which appear in equation (33) are defined as follows:

$$\hat{\mathbf{F}}_{\mathbf{n}}(\hat{\mathbf{W}}) \doteq n_x\hat{\mathbf{F}}_1(\hat{\mathbf{W}})+n_y\hat{\mathbf{F}}_2(\hat{\mathbf{W}}) \tag{63}$$

$$\hat{\mathbf{A}}_{LR}=\begin{pmatrix} 0 & n_{1,LR} & n_{2,LR} & 0 \\ (\hat{c}^2_{LR}-\hat{u}^2_{LR})n_1-\hat{u}_{LR}\hat{v}_{LR}n_2 & 2\hat{u}_{LR}n_1+\hat{v}_{LR}n_{2,LR} & \hat{u}_{LR}n_2 & g\hat{h}_{LR}n_1 \\ (\hat{c}^2_{LR}-\hat{v}^2_{LR})n_2-\hat{u}_{LR}\hat{v}_{LR}n_1 & \hat{v}_{LR}n_1 & \hat{u}_{LR}n_1+2\hat{v}_{LR}n_2 & g\hat{h}_{LR}n_2 \\ \hat{u}_{LR}\hat{f}_{LR}+\hat{v}_{LR}\hat{i}_{LR} & -\hat{f}_{LR} & -\hat{i}_{LR} & 0 \end{pmatrix} \tag{64}$$

and

$$\left|\hat{\mathbf{A}}\left(\hat{\mathbf{W}}\right)\right|=\mathcal{R}\left(\hat{\mathbf{W}}\right)\left|\Lambda\left(\hat{\mathbf{W}}\right)\right|\mathcal{R}^{-1}\left(\hat{\mathbf{W}}\right) \tag{65}$$

where $\Lambda\left(\hat{\mathbf{W}}\right)$ is the diagonal matrix of eigenvalues of $\hat{\mathbf{A}}\left(\hat{\mathbf{W}}\right)$, and $\mathcal{R}\left(\hat{\mathbf{W}}\right)$ is the corresponding right-eigenvectors matrix. The definition of $\hat{f}_{LR}$ and $\hat{i}_{LR}$ is as follows:

$$\begin{aligned} \hat{f}_{LR} &= \frac{\xi A_g\left(\sqrt{\hat{h}_L}+\sqrt{\hat{h}_R}\right)\left(\hat{u}^2_L+\hat{u}_L\hat{u}_R+\hat{u}^2_R+\hat{v}^2_{LR}\right)n_1}{\hat{h}_R\sqrt{\hat{h}_L}+\hat{h}_L\sqrt{\hat{h}_R}} \\ &+ \frac{\xi A_g\left(2\sqrt{\hat{h}_L}\sqrt{\hat{h}_R}\left(\hat{u}_L\hat{v}_L+\hat{u}_R\hat{v}_R\right)+\left(\hat{h}_L\hat{v}_R+\hat{h}_R\hat{v}_L\right)\left(\hat{u}_L+\hat{u}_L\right)\right)n_2}{\left(\hat{h}_R\sqrt{\hat{h}_L}+\hat{h}_L\sqrt{\hat{h}_R}\right)\left(\sqrt{\hat{h}_L}+\sqrt{\hat{h}_R}\right)} \end{aligned} \tag{66}$$

$$\begin{aligned}\hat{i}_{LR} &= \frac{\xi A_g \left(\sqrt{\hat{h}_L}+\sqrt{\hat{h}_R}\right)\left(\hat{v}_L^2+\hat{v}_L\hat{v}_R+\hat{v}_R^2+\hat{u}_{LR}^2\right)n_1}{\hat{h}_R\sqrt{\hat{h}_L}+\hat{h}_L\sqrt{\hat{h}_R}} \\ &+ \frac{\xi A_g \left(2\sqrt{\hat{h}_L}\sqrt{\hat{h}_R}\left(\hat{v}_L\hat{u}_L+\hat{v}_R\hat{u}_R\right)+\left(\hat{h}_L\hat{u}_R+\hat{h}_R\hat{u}_L\right)\left(\hat{v}_L+\hat{v}_L\right)\right)n_2}{\left(\hat{h}_R\sqrt{\hat{h}_L}+\hat{h}_L\sqrt{\hat{h}_R}\right)\left(\sqrt{\hat{h}_L}+\sqrt{\hat{h}_R}\right)}\end{aligned} \tag{67}$$

C Gradient computation

In this section the approximation of the gradients required for the second order extension of the SRNH and MR schemes is addressed. First the gradient approximation for the MR scheme is considered then the SRNH scheme is discussed.

C.1 MR scheme

For the approximation of $\widehat{\nabla\hat{\mathbf{W}}_i}$ appearing in (35) the same technique described in [8] is considered. First, a linear approximation of the gradient in each triangle T_j is considered: defining $tn(j,k), k=1,2,3$ the indexes of the nodes belonging to the triangle T_j, this linear approximation is given by:

$$\widehat{\nabla\hat{\mathbf{W}}}_{|T_j} = \sum_{k=1}^{3} \hat{\mathbf{W}}_{tn(j,k)} \nabla\lambda_j^k \tag{68}$$

where λ_j^k is the barycentric coordinates associated with the k^{th}-vertex. Once $\widehat{\nabla\hat{\mathbf{W}}}_{|T_j}$ is available, a first order approximation of $\widehat{\nabla\hat{\mathbf{W}}_i}$ is the following:

$$\widehat{\nabla\hat{\mathbf{W}}_i^c} \simeq \frac{\sum_{j\in t(i)} |T_j| \widehat{\nabla\hat{\mathbf{W}}}_{|T_j}}{\sum_{j\in t(i)} |T_j|} \tag{69}$$

The proof that (69) is a first order approximation of the gradient of $\hat{\mathbf{W}}$ for regular solutions can be found in [8].
A common characteristic of high order methods for hyperbolic systems is to compute unphysical oscillations near discontinuities. To prevent the generation of those oscillations it is necessary to introduce slope limiters in the reconstructor operator (35). Following [8], the following definition for the slope limiter is considered:

$$\widehat{\nabla\hat{\mathbf{W}}_{i,l}} = \hat{\varphi}_{i,l} \widehat{\nabla\hat{\mathbf{W}}_{i,l}^c} \tag{70}$$

where $\hat{\varphi}_{i,l}$ is the slope limiter function associated with the l^{th}-component of $\widehat{\nabla \hat{\mathbf{W}}}_{ic}$ and is defined as:

$$\left\{\begin{array}{lcl} \hat{\varphi}_{i,l} & = & \min\limits_{j\in N(i)}\{\hat{\varphi}_{ij,l}\} \\ \hat{\varphi}_{ij,l} & = & \max\{0,\min\{r_{ij,l},1\}\} \\ r_{ij,l} & = & \left\{\begin{array}{ll} \dfrac{\hat{W}_{i,l}^{max}-\hat{W}_{i,l}}{\hat{W}_{ij,l}^{*}-\hat{W}_{i,l}} & \text{if} \quad \hat{W}_{ij,l}^{*}-\hat{W}_{i,l}>0 \\ \dfrac{\hat{W}_{i,l}^{min}-\hat{W}_{i,l}}{\hat{W}_{ij,l}^{*}-\hat{W}_{i,l}} & \text{if} \quad \hat{W}_{ij,l}^{*}-\hat{W}_{i,l}<0 \\ 1 & \text{if} \quad \hat{W}_{ij,l}^{*} \end{array}\right. \end{array}\right. \tag{71}$$

where

$$\hat{W}_{i,l}^{min}=\min_{j\in\mathcal{B}_i}\{\hat{W}_{i,l}\}, \quad \hat{W}_{i,l}^{max}=\max_{j\in\mathcal{B}_i}\{\hat{W}_{i,l}\}$$

and

$$\hat{W}_{ij}^{*}=\hat{W}_{ij}^{*}+\widehat{\nabla\hat{\mathbf{W}}_{i}^{c}}\,(\mathbf{c}_{ij}-\mathbf{G}_i) \tag{72}$$

$\mathbf{c}_{ij}$ being the middle point of the segment connecting the i^{th} and j^{th} nodes.

C.2 SRNH scheme

Even if the MUSCL reconstruction operator of the SRNH scheme is different from the MR one an average gradient on the cell i is computed similarly to the MR scheme:

$$\nabla\mathbf{W}_i^c \simeq \frac{\sum\limits_{j\in t(i)}|T_j|\nabla\mathbf{W}_{|T_j}}{\sum\limits_{j\in t(i)}|T_j|} \tag{73}$$

Once $\nabla\mathbf{W}_i^c$ is available the terms $\nabla\mathbf{W}_i\cdot\mathbf{d}_{ij}$ and $\nabla\mathbf{W}_j\cdot\mathbf{d}_{ij}$ in (25) are computed as follows:

$$\left\{\begin{array}{lcl} \nabla\mathbf{W}_i\cdot\mathbf{d}_{ij} & = & \text{minmod}\{\mathbf{W}_j-\mathbf{W}_i, 2\nabla\mathbf{W}_i^c\cdot\mathbf{d}_{ij}-(\mathbf{W}_j-\mathbf{W}_i)\} \\ \nabla\mathbf{W}_j\cdot\mathbf{d}_{ij} & = & \text{minmod}\{\mathbf{W}_j-\mathbf{W}_i, 2\nabla\mathbf{W}_j^c\cdot\mathbf{d}_{ij}-(\mathbf{W}_j-\mathbf{W}_i)\} \end{array}\right. \tag{74}$$

where $\text{minmod}\{a,b\}$ is defined as follows

$$\text{minmod}\{a,b\}=\left\{\begin{array}{ll} \min\{a,b\} & \text{if} a>0, b>0 \\ \max\{a,b\} & \text{if} a<0, b<0 \\ 0 & \text{otherwise} \end{array}\right. \tag{75}$$

References

[1] F. Benkhaldoun, I. Elmahi, and M. Seaïd. Well-balanced finite volume schemes for pollutant transport by shallow water equations on unstructured meshes. *Journal of Computational Physics*, 226(1):180–203, 2007.

[2] F. Benkhaldoun, S. Sahmim, and M. Seaïd. Solution of the Sediment Transport Equations Using a Finite Volume Method Based on Sign Matrix. *SIAM Journal on Scientific Computing*, 31(4):2866–2889, 2009.

[3] F. Benkhaldoun, S. Sahmim, and M. Seaïd. A two-dimensional finite volume morphodynamic model on unstructured triangular grids. *International Journal for Numerical Methods in Fluids*, 63:1296–1327, 2010.

[4] F. Benkhaldoun, S. Sahmim, and M. Seaïd. Mathematical development and verification of a finite volume model for morphodynamic flow applications. *xxx*, xxx(x):xxx, xxx.

[5] A. Bermudez and M.E. Vazquez. Upwind methods for hyperbolic conservation laws with source terms. *Computers & Fluids*, 23(8):1049–1071, 1994.

[6] M.J. Castro Díaz, E.D. Fernández-Nieto, and A.M. Ferreiro. Sediment transport models in Shallow Water equations and numerical approach by high order finite volume methods. *Computers & Fluids*, 37(3):299–316, 2008.

[7] M.J. Castro Díaz, E.D. Fernández-Nieto, A.M. Ferreiro, J.A. García-Rodríguez, and C. Parés. High Order Extensions of Roe Schemes for Two-Dimensional Nonconservative Hyperbolic Systems. *Journal of Scientific Computing*, 39(1):67–114, 2009.

[8] M.J. Castro Díaz, E.D. Fernández-Nieto, A.M. Ferreiro, and C. Parés. Two-dimensional sediment transport models in shallow water equations. A second order finite volume approach on unstructured meshes. *Computer Methods in Applied Mechanics and Engineering*, 198(33–36):2520–2538, 2009.

[9] E. Godlewski and P.A. Raviart. *Numerical approximation of hyperbolic systems of conservation laws*. Number 118 in Applied Mathematical Sciences. Springer, 1996.

[10] S. Gottlieb and C. W. Shu. Total variation diminishing runge-kutta schemes. *Mathematics of Computation*, 67:73–85, 1998.

[11] A.J. Grass. Sediments transport by waves and currents. Report No. FL29, SERC London Cent. Mar. Technol., 1981.

[12] L. Hascoët and V. Pascual. *TAPENADE 2.1 User's Guide*. Technical Report n 300. INRIA, 2004.

[13] J. Hudson and P. K. Sweby. Formulations for Numerically Approximating Hyperbolic Systems Governing Sediment Transport. *Journal of Scientific Computing*, 19(1–3):225–252, 2003.

[14] R. Martin and H. Guillard. A second order defect correction scheme for unsteady problems. *Computers & Fluids*, 25(1):9–27, 1996.

[15] E. Sinibaldi, F. Beux, M. Bilanceri, and M. V. Salvetti. *A Second-Order Linearised Implicit Formulation for Hyperbolic Conservation Laws, with Application to Barotropic Flows*. ADIA 2008-4. University of Pisa, June 2008.

[16] B. van Leer. Towards the ultimate conservative difference scheme V: a second-order sequel to Godunov's method. *Journal of Computational Physics*, 32(1):101–136, 1979.

Centre de recherche INRIA Sophia Antipolis – Méditerranée
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399